Подписано электронной подписью:
Вержицкий Данил Григорьевич
Должность: Директор КГПИ ФГБОУ ВО «КемГУ»
Дата и время: 2024-02-21 00:00:00
471086fad29a3b30e244c728abc3661ab35e9d50310dcf0e75e03a5b6fdf6436

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кемеровский государственный университет»
Кузбасский гуманитарно-педагогический институт
федерального государственного бюджетного образовательного учреждения
высшего образования
«Кемеровский государственный университет»
Факультет информатики, математики и экономики

УТВЕРЖДАЮ
Декан
А. В. Фомина
10 февраля 2022 г.

# Рабочая программа дисциплины

Б1.В.02 Параллельные и распределенные вычислительные системы
*Код, название дисциплины*

Направление подготовки
02.03.03 Математическое обеспечение и администрирование информационных систем
*Код, название направления*

Направленность (профиль) подготовки
Программное и математическое обеспечение информационных систем

Программа бакалавриата

Квалификация выпускника
*бакалавр*

Форма обучения
*Очная*

Год набора 2021

Новокузнецк 2022

# Оглавление

1 Цель дисциплины. ....3
1.1 Формируемые компетенции ....3
1.2 Индикаторы достижения компетенций ....3
1.3 Знания, умения, навыки (ЗУВ) по дисциплине ....3
2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации. ....4
3. Учебно-тематический план и содержание дисциплины. ....5
3.1 Учебно-тематический план ....5
3.2. Содержание занятий по видам учебной работы ....6
4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации. ....8
5 Материально-техническое, программное и учебно-методическое обеспечение дисциплины. ....9
5.1 Учебная литература ....9
5.2 Материально-техническое и программное обеспечение дисциплины. ....10
5.3 Современные профессиональные базы данных и информационные справочные системы. ....11
6 Иные сведения и (или) материалы. ....11
6.1.Примерные темы письменных учебных работ ....11
6.2. Примерные вопросы и задания / задачи для промежуточной аттестации ....15

## 1 Цель дисциплины.

В результате освоения данной дисциплины у обучающегося должны быть сформированы компетенции основной профессиональной образовательной программы бакалавриата (далее - ОПОП):

ПК-2.

Содержание компетенций как планируемых результатов обучения по дисциплине см. таблицы 1 и 2.

## 1.1 Формируемые компетенции

Таблица 1 - Формируемые дисциплиной компетенции

| Наименование вида компетенции *(универсальная, общепрофессиональная, профессиональная)* | Наименование категории (группы) компетенций | Код и название компетенции |
|---|---|---|
| профессиональная | | ПК-2 Способен определять структуры данных, а также технологии обработки и доступа к данным каждого компонента и программного средства в целом |

## 1.2 Индикаторы достижения компетенций

Таблица 2 – Индикаторы достижения компетенций, формируемые дисциплиной

| Код и название компетенции | Индикаторы достижения компетенции по ОПОП | Дисциплины и практики, формирующие компетенцию ОПОП |
|---|---|---|
| ПК-2 Способен определять структуры данных, а также технологии обработки и доступа к данным каждого компонента и программного средства в целом | ПК 2.1 Определяет входные-выходные данные и их взаимосвязи для каждого компонента и программного средства в целом.<br><br>ПК 2.2 Определяет структуры данных и алгоритмы каждого компонента и программного средства в целом.<br><br>ПК 2.3 Использует различные технологии обработки данных в программном средстве.<br><br>ПК 2.4 Определяет перечень возможных технологий доступа к данным. | Б1.В.02 Параллельные и распределенные вычислительные системы<br>Б1.В.03 Современные технологии программирования SQL<br>Б1.В.07 Структуры и алгоритмы компьютерной обработки данных<br>Б2.В.01(Пд) Преддипломная практика<br>Б3.01(Д) Подготовка к процедуре защиты и защита выпускной квалификационной работы |

## 1.3 Знания, умения, навыки (ЗУВ) по дисциплине

Таблица 3 – Знания, умения, навыки, формируемые дисциплиной

| Код и название компетенции | Индикаторы достижения компетенции, закрепленные за дисциплиной | Знания, умения, навыки (ЗУВ), формируемые дисциплиной |
|---|---|---|
| ПК-2 Способен определять структуры | ПК 2.1 Определяет входные-выходные данные и их | Знать:<br>компоненты программно-технических |

| Код и название компетенции | Индикаторы достижения компетенции, закрепленные за дисциплиной | Знания, умения, навыки (ЗУВ), формируемые дисциплиной |
|---|---|---|
| данных, а также технологии обработки и доступа к данным каждого компонента и программного средства в целом | взаимосвязи для каждого компонента и программного средства в целом.<br><br>ПК 2.2 Определяет структуры данных и алгоритмы каждого компонента и программного средства в целом.<br><br>ПК 2.3 Использует различные технологии обработки данных в программном средстве.<br><br>ПК 2.4 Определяет перечень возможных технологий доступа к данным. | архитектур параллельных вычислительных систем; виды параллелизма, уровни распараллеливания; структуры данных и модели параллельной программы для вычислительной системы с распределённой памятью; технологии доступа данных в параллельных системах; основы проектирования, построения и функционирования распределенных систем.<br>Уметь:<br>применять декомпозицию, проектирование взаимодействий, укрупнение и планирование вычислений при разработке параллельного алгоритма и программных средств; выявлять информационные зависимости между итерациями циклических участков программы; самостоятельно находить алгоритмы решения задач, требующихся для проектирования, построения и использования распределенных систем, в том числе нестандартных и проводить их анализ.<br>Владеть:<br>способами преобразования циклов для ликвидации информационных зависимостей между итерациями; навыками разработки, компиляции и отладки параллельных программ; навыками определения алгоритмов компонент параллельных программ и программных средств; навыками освоения большого объема информации и решения задач распределенных систем. |

## 2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации.

Таблица 4 – Объем и трудоемкость дисциплины по видам учебных занятий

| Общая трудоемкость и виды учебной работы по дисциплине, проводимые в разных формах | Объём часов по |
|---|---|

| | формам обучения |
|---|---|
| | ОФО |
| 1 Общая трудоемкость дисциплины | 180 |
| 2 Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего) | 48 |
| Аудиторная работа (всего): | 48 |
| в том числе: | |
| лекции | 18 |
| практические занятия, семинары | 30 |
| практикумы | |
| лабораторные работы | |
| в интерактивной форме | |
| в электронной форме | |
| Внеаудиторная работа (всего): | |
| в том числе, индивидуальная работа обучающихся с преподавателем | |
| подготовка курсовой работы /контактная работа | |
| групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие групповую или индивидуальную работу обучающихся с преподавателем*)* | |
| творческая работа (эссе) | |
| 3 Самостоятельная работа обучающихся (всего) | 96 |
| 4 Промежуточная аттестация обучающегося: | Экзамен – 5 семестр (36 часов) |

## 3. Учебно-тематический план и содержание дисциплины.

### 3.1 Учебно-тематический план

Таблица 5 - Учебно-тематический план очной формы обучения

| № недели п/п | Разделы и темы дисциплины по занятиям | Общая трудоёмкость *(всего час.)* | Трудоемкость занятий (час.) | | | Формы текущего контроля и промежуточной аттестации успеваемости |
|---|---|---|---|---|---|---|
| | | | ОФО | | | |
| | | | Аудиторн. Занятия | | СРС | |
| | | | лекц. | практ. | | |
| **Семестр 5** | | | | | | |
| | *1. Понятие параллельных и распределенных систем* | 10 | 2 | 2 | 6 | |
| 1 | 1.1 Определение и особенности распределённых систем | 3 | 1 | 1 | 1 | |
| 2 | 1.2 Архитектура параллельных и распределенных систем | 7 | 1 | 2 | 4 | Опрос №1 |
| | *2. Параллельное программирование* | 40 | 4 | 9 | 27 | |
| 3-5 | 2.1 Параллельные вычисления | 13 | 2 | 3 | 8 | Индивидуальное задание №1 |
| 4-6 | 2.2 Технология параллельного программирования систем с общей памятью на OpenMP | 10 | - | 4 | 6 | Опрос №2 |

| № недели п/п | Разделы и темы дисциплины по занятиям | Общая трудоёмкость *(всего час.)* | Трудоемкость занятий (час.) | | | Формы текущего контроля и промежуточной аттестации успеваемости |
|---|---|---|---|---|---|---|
| | | | ОФО | | | |
| | | | Аудиторн. Занятия | | СРС | |
| | | | лекц. | практ. | | |
| **Семестр 5** | | | | | | |
| 6 | 2.3 Использование графических процессоров | 7 | 2 | - | 5 | Опрос №3 |
| 7-9 | 2.4 Интерфейс передачи сообщений MPI | 10 | - | 3 | 7 | Опрос №4 |
| | *3. Модели распределенных систем* | 36 | 6 | 11 | 19 | |
| 7-11 | 3.1 Модель распределенного исполнения | 12 | 2 | 3 | 7 | Индивидуальное задание №2 |
| 9-12 | 3.2 Логическое время | 4 | - | 2 | 2 | |
| 10-13 | 3.3 Синхронное и асинхронное исполнение | 6 | 2 | 2 | 2 | |
| 11-14 | 3.4 Модели отказов | 4 | - | 2 | 2 | |
| 12-15 | 3.5 Глобальное состояние | 10 | 2 | 2 | 6 | Индивидуальное задание №3 |
| | *4. Распределенные системы* | 58 | 6 | 6 | 46 | |
| 13 | 4.1 Коммуникационная подсистема | 4 | - | - | 4 | |
| 14-16 | 4.2 Синхронизация | 14 | 1 | 2 | 11 | Индивидуальное задание №4 |
| 15 | 4.3 Репликация и консистентность | 4 | 1 | - | 3 | |
| 16 | 4.4 Безопасность | 12 | - | - | 12 | Индивидуальное задание №5 |
| 17 | 4.5 Системы хранения данных | 14 | 2 | 2 | 10 | Индивидуальное задание №6 |
| 18 | 4.6 Распределенные вычисления | 10 | 2 | 2 | 6 | |
| 18 | Промежуточная аттестация | 36 | | | | экзамен |
| | Всего: | 216 | 18 | 30 | 96 | |

## 3.2. Содержание занятий по видам учебной работы

Таблица 6 – Содержание дисциплины

| № п/п | Наименование раздела, темы дисциплины | Содержание занятия |
|---|---|---|
| | **Семестр 5** | |
| *Содержание лекционного курса* | | |
| 1 | *Понятие параллельных и распределенных систем* | |
| 1.1 | Определение и особенности распределённых систем | *Определение распределенной системы. Особенности распределенных систем: отсутствие общей памяти, отсутствие общих физических часов, асинхронная связь и асинхронное исполнение, географическая удаленность, автономность и гетерогенность, отказоустойчивость, недетерминизм. Целесообразность построения* |

| № п/п | Наименование раздела, темы дисциплины | Содержание занятия |
|---|---|---|
| | | *распределенных систем. Примеры. Применение. Параллельные и распределенные системы.* |
| 1.2 | Архитектура параллельных и распределенных систем | *Сервисы, роли и архитектурные стили распределенных систем. Клиент-сервер, одноранговые сети, сервисно-ориентированная архитектура. Определение параллельной и распределенной системы. Параллельная архитектура: ОКОД, ОКМД, МКОД, МКМД.* |
| *2.* | *Параллельное программирование* | |
| 2.1 | Параллельные вычисления | *Внутренний параллелизм. Распараллеливание циклов. Преобразование циклов. Проблемы разработки параллельных программ.* |
| *3.* | *Модели распределенных систем* | |
| 3.1 | Модель распределенного исполнения | *Общее описание модели. Модель коммуникационного канала. Событийное описание. Упорядочивание событий. Отношение причинного предшествования* |
| 3.3 | Синхронное и асинхронное исполнение | *Асинхронное исполнение. Синхронное исполнение. Эмуляции синхронных систем асинхронными и наоборот. Эмуляции* |
| 3.5 | Глобальное состояние | *Распределенная сборка мусора. Распределенное обнаружение тупиков. Распределенное обнаружение завершения. Фиксация глобального состояния* |
| *4.* | *Распределенные системы* | |
| 4.2 | Синхронизация | *Алгоритмы синхронизации часов. Алгоритмы выбора. Распределенное взаимоисключение. Консенсус. Распределенное взаимоисключение* |
| 4.3 | Репликация и консистентность | *Модель и архитектура управления реплицированными данными. Пассивная и активная репликации. Отказоустойчивость сервиса репликации. Модели консистентности. Размещение и обновление реплик* |
| 4.5 | Системы хранения данных | *Современные подходы к построению систем распределенного хранения данных. Распределенные кластерные файловые системы. Пиринговые системы. Масштабируемость P2P-систем* |
| 4.6 | Распределенные вычисления | *Влияние аппаратной архитектуры сети на производительность. Влияние решаемой задачи на производительность. MPI. Распараллеливание задач по вычислительным ресурсам. Сильно связанные задачи и слабо связанные задачи. Модели распределенных вычислений. Общая структура функционирования. Компоненты метасистемы* |
| *Содержание практических занятий* | | |
| 1 | *Понятие параллельных и распределенных систем* | |
| 1.1 | Особенности применения распределенных систем | *Целесообразность построения распределенных систем.* |
| 1.2 | Моделирование архитектура параллельных | *Моделирование распределённых систем и распределенных процессов.* |

| № п/п | Наименование раздела, темы дисциплины | Содержание занятия |
|---|---|---|
| | и распределенных систем | |
| 2 | *Параллельное программирование* | |
| 2.1 | Реализация параллельных вычисления | *Реализация параллельных циклов, преобразование циклов.* |
| 2.2 | Применение технологии параллельного программирования систем с общей памятью на OpenMP | *Директивы OpenMP. Библиотека функций OpenMP* |
| 2.4 | Использование интерфейса передачи сообщений MPI | *Функции инициализации, завершения, определения окружения. Обмен сообщения типа «точка-точка». Управление группами ветвей и коммуникаторами. Динамическое порождение ветвей. Параллельный ввод/вывод* |
| *3.* | *Модели распределенных систем* | |
| 3.1 | Построение модели распределенного исполнения | *Моделирование распределённых процессов.* |
| 3.2 | Реализация алгоритмов. Логическое время | *Использование логического времени. Скалярные и векторные часы.* |
| 3.3 | Использование синхронных и асинхронных примитивов обмена сообщениями | *Реализация синхронных и асинхронных примитивов. Синхронизаторы* |
| 3.4 | Построение моделей отказов | *Построение модели отказов и неисправности* |
| 3.5 | Построение алгоритмов обнаружения завершения | *Алгоритм обнаружения завершения* |
| *4.* | *Распределенные системы* | |
| 4.2 | Реализация синхронизация | *Алгоритмы взаимоисключения* |
| 4.5 | Распределенное хранение | *Синхронизация больших объектов* |
| 4.6 | Распределенные вычисления | *Сравнение моделей распределенных вычислений. Распараллеливание задач.* |
| | Промежуточная аттестация – *экзамен* | |

## 4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации.

Для положительной оценки по результатам освоения дисциплины обучающемуся необходимо выполнить все установленные виды учебной работы. Оценка результатов работы обучающегося в баллах (по видам) приведена в таблице 7.

Таблица 7 - Шкала и показатели оценивания результатов учебной работы обучающихся по видам в балльно-рейтинговой системе (БРС)

| Учебная работа (виды) | Сумма баллов | Виды и результаты учебной работы | Оценка в аттестации (шкала и показатели оценивания) | Баллы (17 недель) |
|---|---|---|---|---|

| | | | | |
|---|---|---|---|---|
| Текущая учебная работа в семестре (Посещение занятий по расписанию и выполнение заданий) | **60** | Лекционные занятия (конспект) (9 занятий) | **1 балл** посещение 1 лекционного занятия | 0 - 8 |
| | | Практические работы (отчет о выполнении практической работы) (18 работ). | **0,5 балла** - посещение 1 практического занятия и выполнение работы на 51-65% **1 балл** – посещение 1 занятия и существенный вклад на занятии в работу всей группы, самостоятельность и выполнение работы на 85,1-100% | 9 – 18 |
| | | Индивидуальные задания (отчет о выполнении индивидуальных заданий) (6 работ) | **За одно ИЗ :** **3 баллов** (выполнено 51 - 65% заданий) **4 балла** (выполнено 66 - 85% заданий) **5 балла** (выполнено 86 - 100% заданий) | 18 - 30 |
| | | Опросы (4 опроса) | **0,5 балла** (пороговое значение) **1 балл** (максимальное значение) | 2 - 4 |
| **Итого по текущей работе в семестре** | | | | 31 - 60 |
| Промежуточная аттестация (экзамен) | 40 | Тест. | **5 баллов** (пороговое значение) **10 баллов** (максимальное значение) | 5 - 10 |
| | | Решение задачи 1. | **7 баллов** (пороговое значение) **15 баллов** (максимальное значение) | 7 - 15 |
| | | Решение задачи 2. | **8 баллов** (пороговое значение) **15 баллов** (максимальное значение) | 8 - 15 |
| **Итого по промежуточной аттестации (экзамену)** | | | | 20 – 40 б. |
| **Суммарная оценка по дисциплине:** Сумма баллов текущей и промежуточной аттестации | | | | 51 – 100 б. |

В промежуточной аттестации оценка выставляется в ведомость в 100-балльной шкале и в буквенном эквиваленте (таблица 8)

Таблица 8 – Соотнесение 100-балльной шкалы и буквенного эквивалента оценки

| *Сумма набранных баллов* | Уровни освоения дисциплины и компетенций | *Экзамен* | | *Зачет* |
|---|---|---|---|---|
| | | *Оценка* | *Буквенный эквивалент* | *Буквенный эквивалент* |
| 86 - 100 | Продвинутый | 5 | отлично | Зачтено |
| 66 - 85 | Повышенный | 4 | хорошо | |
| 51 - 65 | Пороговый | 3 | удовлетворительно | |
| 0 - 50 | Первый | 2 | неудовлетворительно | Не зачтено |

# 5 Материально-техническое, программное и учебно-методическое обеспечение дисциплины.

## 5.1 Учебная литература

### Основная учебная литература

Малявко, А. А. Параллельное программирование на основе технологий OpenMP, MPI, CUDA : учебное пособие для академического бакалавриата / А. А. Малявко. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2019. — 129 с. — (Высшее образование). — ISBN 978-5-534-11827-8. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://urait.ru/bcode/446247

Бабичев, С. Л. Распределенные системы : учебное пособие для вузов / С. Л. Бабичев, К. А. Коньков. — Москва : Издательство Юрайт, 2020. — 507 с. —

(Высшее образование). — ISBN 978-5-534-11380-8. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://urait.ru/bcode/457005

**Дополнительная учебная литература**

Трофимов, В. В. Алгоритмизация и программирование : учебник для вузов / В. В. Трофимов, Т. А. Павловская ; под редакцией В. В. Трофимова. — Москва : Издательство Юрайт, 2020. — 137 с. — (Высшее образование). — ISBN 978-5-534-07834-3. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://urait.ru/bcode/452333

Стасышин, В. М. Базы данных: технологии доступа : учебное пособие для вузов / В. М. Стасышин, Т. Л. Стасышина. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2020. — 164 с. — (Высшее образование). — ISBN 978-5-534-08687-4. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://urait.ru/bcode/463499

Замятина, О. М. Вычислительные системы, сети и телекоммуникации. Моделирование сетей : учебное пособие для вузов / О. М. Замятина. — Москва : Издательство Юрайт, 2020. — 159 с. — (Высшее образование). — ISBN 978-5-534-00335-2. — Текст : электронный // ЭБС Юрайт [сайт]. — URL: https://urait.ru/bcode/451319

## 5.2 Материально-техническое и программное обеспечение дисциплины.

Учебные занятия по дисциплине проводятся в учебных аудиториях КГПИ КемГУ:

| | |
|---|---|
| **615** Учебная аудитория (мультимедийная) для проведения:<br>- занятий лекционного типа.<br>**Специализированная (учебная) мебель:** доска меловая, кафедра, столы, стулья.<br>**Оборудование для презентации учебного материала:** *стационарное* - компьютер, экран, проектор, акустическая система (колонки).<br>**Используемое программное обеспечение:** Ubuntu Linux(свободно распространяемое ПО), LibreOffice (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО).<br>**Интернет с обеспечением доступа в ЭИОС.** | 654079, Кемеровская область, г. Новокузнецк, пр-кт Металлургов, д. 19 |
| **502 Лаборатория компьютерного моделирования.** Учебная аудитория (мультимедийная) для проведения:<br>- занятий лабораторного типа;<br>- групповых и индивидуальных консультаций;<br>- самостоятельной работы;<br>- текущего контроля и промежуточной аттестации.<br>**Специализированная (учебная) мебель:** доска меловая, столы компьютерные, стулья.<br>**Оборудование для презентации учебного материала:** *стационарное* -компьютер, экран, проектор, наушники.<br>**Лабораторное оборудование**: *стационарное* – компьютеры для обучающихся (16 шт.). | 654079, Кемеровская область, г. Новокузнецк, пр-кт Металлургов, д. 19 |

| | |
|---|---|
| **Используемое программное обеспечение:** MS Windows (Microsoft Imagine Premium 3 year по сублицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.), LibreOffice (свободно распространяемое ПО), Firefox 14 (свободно распространяемое ПО), Яндекс.Браузер (отечественное свободно распространяемое ПО), Netbeans IDE 7.0.1 для Firefox (свободно распространяемое ПО), Opera 12 (свободно распространяемое ПО), Консультант Плюс (отечественное ПО, договор об инфо поддержке 1.04.2007), Microsoft Visual Studio (Microsoft Imagine Premium 3 year по сублицензионному договору № 1212/КМР от 12.12.2018 г. до 12.12.2021 г.).<br>**Интернет с обеспечением доступа в ЭИОС.** | |

## 5.3 Современные профессиональные базы данных и информационные справочные системы.

**Перечень СПБД и ИСС по дисциплине**

1. CITForum.ru - on-line библиотека свободно доступных материалов по информационным технологиям на русском языке - http://citforum.ru
2. Научная электронная библиотека eLIBRARY.RU – крупнейший российский информационный портал в области науки, технологии, медицины и образования, содержащий рефераты и полные тексты - www.elibrary.ru
3. Единое окно доступа к образовательным ресурсам - http://window.edu.ru/

# 6 Иные сведения и (или) материалы.

## 6.1.Примерные темы письменных учебных работ

**Темы опроса**

Опрос №1

Исполнение нескольких распределенных процессов

Клиент-серверная архитектура

Параллельная система

Распределенная система

Классификация параллельных систем

ОКОД

ОКМД

Архитектура МКМД

Опрос №2

1. Перечислите составные части технологии OpenMP.
2. С помощью какой директивы (директив) создаются новые параллельные области программы?
3. Что такое критическая секция программы?
4. Каким образом можно установить нужное количество потоков для создания

очередной параллельной области?

5. Как обеспечить выполнение фрагмента параллельной области только главным потоком?

6. Какая опция директивы OpenMP for используется для указания способа распределения итераций цикла между потоками параллельной области?

7. Что такое deadlock? Каким правилам нужно следовать, чтобы избежать возможности попадания параллельной программы в deadlock?

8. Что такое сведение данных? Какие опции и в каких директивах используются для выполнения сведения?

9. Что делает директива OpenMP threadprivate?

10. Как обеспечить выполнение фрагмента параллельной области потоком с максимальным номером в данной параллельной области?

11. Для чего используется опция firstprivate? Чем она отличается от опций private и lastprivate?

12. Что такое вложенная параллельная область программы? В каких случаях ее нельзя создать?

13. Что такое неявная барьерная синхронизация? С помощью каких средств ее можно отменить?

14. Для чего используются директивы OpenMP sections и section? Что делает каждая из этих директив?

15. Перечислите все средства синхронизации потоков в OpenMP.

16. Перечислите возможные способы распределения итераций цикла между потоками.

17. Что делает директива OpenMP atomic?

18. В чем состоит различие между общими и локальными переменными потока?

19. С помощью каких средств можно ограничить глубину вложенности параллельных областей программы?

20. От чего зависит равномерность загрузки процессоров/ядер системы с общей памятью?

21. Каким образом функция, вызываемая из параллельной программы, может выяснить, в последовательной или параллельной области она выполняется?

22. Как обеспечить выполнение фрагмента параллельной области в точности одним потоком?

23. Какое значение будет иметь переменная count в результате выполнения фрагмента параллельной программы:

```
int count = 0;
#pragma omp parallel for
for(int i = 0; i < 10; i++){
count++;
}
```

Опрос №3

1. Что такое ядро в терминологии CUDA? 2. Как задается размерность и количество потоков ядра, выполняемого графическим процессором?

3. Перечислите виды и характеристики памяти, доступной из программы GPU.

4. Как указать требуемое размещение переменной в памяти GPU (регистровой,

разделяемой, глобальной, памяти текстур,…)?

5. Перечислите виды и характеристики памяти, доступной из программы CPU при использовании архитектуры CUDA.

6. Перечислите встроенные векторные типы данных расширения языка C и объясните смысл их наименований.

7. Как в программе для CPU обеспечить синхронизацию с программой для GPU?

8. Если к одному CPU подключено несколько видеокарт NVidia архитектуры CUDA, то каким образом можно обеспечить их одновременную загрузку?

9. Перечислите виды и характеристики памяти GPU, доступной из программы основного процессора.

10. Какие ограничения наложены на функции, которые должны выполняться на GPU?

11. Что такое мультипроцессор в терминологии CUDA?

12. Какие ограничения наложены на функции, выполняемые в графическом процессоре?

13. Как определить версию и технические характеристики графического процессора NVIDIA?

14. Перечислите и охарактеризуйте группы библиотечных функций Run-time библиотеки CUDA.

15. Что такое сетка, блок, поток в терминологии CUDA?

16. Что такое тип данных dim3?

17. Какова максимальная размерность сетки в блоках?

18. Перечислите и охарактеризуйте функции компонентов программного обеспечения CUDA.

19. Какую модель параллелизма реализует архитектура CUDA?

20. Что делает препроцессор nvcc?

21. Можно ли получить указатель на функцию, выполняемую графическим процессором?

Опрос №4

1. Какие виды виртуальных топологий можно реализовывать с использованием MPI?

2. Что такое барьерная синхронизация в MPI? Какие еще виды синхронизации существуют в стандарте MPI-1?

3. Перечислите основные группы функций стандарта MPI-1.

4. Может ли MPI-программа, выполняющаяся на многоядерном узле, использовать все его ядра? И если да, то каким образом?

5. Что такое пользовательская операция редукции?

6. Чем различаются размер и протяженность типов данных MPI?

7. Что такое буферизованная передача сообщений?

8. Что такое коммуникатор? Какие проблемы решаются с использованием этого понятия?

9. Сколько в MPI операций приема сообщений типа мточка-точкаЋ? Перечислите и охарактеризуйте их.

10. Что такое коллективное взаимодействие ветвей программы?

11. Как путем вызова одной функции можно одновременно передать и принять

блок данных?

12. Перечислите предопределенные операции редукции данных MPI.

13. Что такое виртуальная топология? Как она связана с физической топологией вычислительной сети/кластера/комплекса?

14. Для чего в MPI-программах можно использовать производные типы данных?

15. Каково назначение библиотеки MPE?

16. Что понимается под операциями сдвига и циклического сдвига данных? Какие функции обычно используются для их реализации?

17. Что такое исключающая редукция (MPI_Exscan), как она выполняется?

18. Может ли ветвь параллельной программы с помощью MPI передавать сообщения сама себе?

19. Какие операции рассылки и сбора данных реализованы в MPI?

20. Должны ли абсолютно все ветви параллельной программы принимать участие в коллективном взаимодействии?

21. Сколько конструкторов производных типов реализовано в MPI? Перечислите и охарактеризуйте их.

22. Чем различаются топологии мторЋ и мрешеткаЋ?

23. Что такое вид файла? Для чего используется это понятие?

24. Какие способы синхронизации используются при удаленном доступе к памяти?

25. Что такое интер-коммуникатор? Чем это понятие отличается от понятия интра-коммуникатора?

26. Что такое файловый тип?

27. Что такое мокноЋ при удаленном доступе памяти? Как создать окно?

28. Что такое синхронизация с блокировками при удаленном доступе к памяти? Какие еще существуют виды синхронизации?

29. Что такое смещение файла? Чем оно отличается от типового смещения?

30. Какие функции могут быть использованы для модификации состояния памяти мокнаЋ другой ветви?

31. Перечислите основные группы функций библиотеки MPE.

32. Чем различаются индивидуальные и общие указатели файлов?

33. Что такое е-тип? Может ли в качестве е-типа использоваться базовый тип?

34. Что такое заборная синхронизация?

35. Какие типы доступа к файлам существуют в MPI?

36. С помощью каких функций выполняется создание новых групп ветвей?

37. Что такое синхронизация с пассивным адресатом?

38. Что такое доступ к файлу по явному смещению? Являются ли операции такого доступа коллективными?

39. Можно ли изменять вид файла в процессе выполнения программы?

**Темы индивидуального задания**

Индивидуальное задание №1

Реализовать алгоритмы работы с массивами, используя для циклов блочное, циклическое и блочно-циклическое распределения. При оформлении работы, предоставить блок схемы и код исходного и распараллеленного алгоритмов, сравнение

различных подходов к распределению. Указать методы, используемые для преобразования циклов.

Индивидуальное задание №2

Реализовать модель асинхронных распределенных систем на любом языке программирования, допускающем использование параллельно исполняющихся процедур.

Индивидуальное задание №3

1. Покажите, что любое базовое вычисление можно модифицировать так, что оно будет состоять только из событий трех групп, Send, Receive и Internal.

2. Напишите какой-либо алгоритм Announce извещения о завершении вычисления.

Индивидуальное задание №4

Реализуйте распределенный алгоритм взаимоисключения в виде рабочей функции

Индивидуальное задание №5

Спроектировать систему, которая удовлетворяет следующим требованиям:

- не должно быть централизованной базы данных пользователей;
- общие данные делятся на фрагменты фиксированной длины, которые мы будем называть чанками, от английского слова chunk — кусок;
- каждый чанк данных может быть доступен тем пользователям, которым владелец разрешил доступ;
- коммуникации между процессами проходят в небезопасной среде. Любой посторонний пользователь может читать все сообщений между процессами;
- каждый чанк может быть сохранен отдельно от других, процесс, создавший чанк может прекратить свое существование.

Индивидуальное задание №6

1. Реализуйте простой вариант алгоритма синхронизации больших данных.

2. Реализуйте аддитивную хэш-функцию с помощью продвинутого алгоритма.

## 6.2. Примерные вопросы и задания / задачи для промежуточной аттестации

*Семестр 8*

**Таблица 9 - Примерные теоретические вопросы и практические задания к экзамену**

| Разделы и темы | Примерные теоретические вопросы | Примерные практические задания |
|---|---|---|
| *1. Понятие параллельных и распределенных систем* | | |
| 1.1 Определение и особенности распределённых систем | 1. Особенности распределенных систем<br>2. Определение распределенной системы<br>Отказоустойчивость в распределенных системах | |
| 1.2 Архитектура параллельных и | 3. Архитектура МКМД<br>4. Архитектура ОКОД | |

| | | |
|---|---|---|
| распределенных систем | 5. Сервисы распределенных систем<br>6. Архитектурные стили распределенных систем | |
| *2. Параллельное программирование* | | |
| 2.1 Параллельные вычисления | 7. Блочное распараллеливание циклов<br>9. Развертка циклов<br>10. Виды преобразования циклов | *Реализовать блочное распараллеливание циклов*<br>*Реализовать преобразование циклов расширением скаляра* |
| 2.2 Технология параллельного программирования систем с общей памятью на OpenMP | 11. Средства синхронизации потоков в OpenMD<br>12. Неявная барьерная синхронизация<br>13. Общие и локальные переменные потока | *Используя OpenMP, реализовать параллельный алгоритм* |
| 2.3 Использование графических процессоров | 14. Сетка, блок и поток<br>15. Ограничения на функции, исполняемые на GPU | |
| 2.4 Интерфейс передачи сообщений MPI | 16. Буферизированная передача сообщений<br>17. Коллективное взаимодействие ветвей программы<br>18. Исключающая редукция | *Реализовать параллельный ввод* |
| *3. Модели распределенных систем* | | |
| 3.1 Модель распределенного исполнения | 19. Модель коммуникационного канала<br>20. Упорядочивание событий | |
| 3.2 Логическое время | 21. Логические часы<br>22. Скалярное время<br>23. Векторное время | *Реализовать алгоритм определения времени событий* |
| 3.3 Синхронное и асинхронное исполнение | 24. Асинхронное исполнение<br>25. Эмуляции | |
| 3.4 Модели отказов | 26. Отказы процессов<br>27. Иерархия моделей неисправности | *Построение модели отказов* |
| 3.5 Глобальное состояние | 28. Распределенная сборка мусора<br>29. Распределенное обнаружение завершения<br>30. Фиксация глобального состояния | |
| *4. Распределенные системы* | | |
| 4.1 Коммуникационная подсистема | 31. Алгоритмы на графах<br>32. Удаленные вызовы | |
| 4.2 Синхронизация | 33. Алгоритмы синхронизации часов<br>34. Распределенные взаимоисключения | *Реализация алгоритма синхронизации* |
| 4.3 Репликация и консистентность | 35. Пассивная репликация<br>36. Активная репликация<br>37. Размещение и обновление реплик | |
| 4.4 Безопасность | 38. Модели контроля доступа<br>39. Дискретное управление | |

| | | |
|---|---|---|
| | доступом | |
| 4.5 Системы хранения данных | 40. Распределенные кластерные файловые системы<br>41. Пиринговые системы | |
| 4.6 Распределенные вычисления | 42. Модели распределённых вычислений.<br>43. Компоненты метасистемы | |

Составитель (и): Штейнбрехер О.А., канд. техн. наук, доцент кафедры ИВТ
(*фамилия, инициалы и должность преподавателя (ей)*)