Подписано электронной подписью:
Вержицкий Данил Григорьевич
Должность: Директор КГПИ КемГУ
Дата и время: 2025-04-23 00:00:00
471086fad29a3b30e244c728abc3661ab35e9d50310dcf0e75e03a5b6fdf6436

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кемеровский государственный университет»
Новокузнецкий институт (филиал)
федерального государственного бюджетного образовательного учреждения
высшего образования
«Кемеровский государственный университет»

(Наименование филиала, где реализуется данная дисциплина)

Факультет информатики, математики и экономики

УТВЕРЖДАЮ
Декан А.В. Фомина
« 09 » февраля 2023 г.

Рабочая программа дисциплины

К.М.04.07 Параллельные и распределенные вычислительные системы

Код, название дисциплины

Направление подготовки
09.03.01 Информатика и вычислительная техника

Код, название направления

Направленность (профиль) подготовки
Автоматизированные системы обработки информации и управления

Программа бакалавриата

Квалификация выпускника
бакалавр

Форма обучения
Заочная

Год набора 2023

Новокузнецк 2023

Содержание

1 Цель дисциплины. 3

1.1 Формируемые компетенции, индикаторы достижения компетенций, знания, умения, навыки 3

1.2. Место дисциплины 4

2 Объём и трудоёмкость дисциплины по видам учебных занятий. Формы промежуточной аттестации. 4

3. Учебно-тематический план и содержание дисциплины. 4

4 Порядок оценивания успеваемости и сформированности компетенций обучающегося в текущей и промежуточной аттестации. 5

5 Учебно-методическое обеспечение дисциплины. 6

5.1 Учебная литература 6

5.2 Программное и информационное обеспечение освоения дисциплины. 7

5.2.1 Программное обеспечение 7

5.3.2 Современные профессиональные базы данных и информационные справочные системы. 8

6 Иные сведения и (или) материалы 8

6.1.Примерные темы письменных учебных работ 8

6.2. Примерные вопросы и задания / задачи для промежуточной аттестации 12

1 Цель дисциплины.

В результате освоения данной дисциплины у обучающегося должны быть сформированы компетенции основной профессиональной образовательной программы бакалавриата (далее - ОПОП): ОПК-2; ОПК-9

1.1 Формируемые компетенции, индикаторы достижения компетенций, знания, умения, навыки

Таблица 1 – Индикаторы достижения компетенций, формируемые дисциплиной

Код и название компетенции	Индикаторы достижения компетенции, закрепленные за дисциплиной	Знания, умения, навыки (ЗУВ), формируемые дисциплиной
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;	ОПК-2.1. Выбирает современные информационные технологии, в том числе отечественного производства, и программные средства для решения поставленной задачи профессиональной деятельности. ОПК-2.2. Применяет технологии параллельного программирования в соответствии с заданием	Знать: – компоненты программно-технических архитектур параллельных вычислительных систем; – виды параллелизма, уровни распараллеливания; – модель параллельной программы для вычислительной системы с распределённой памятью; – основы проектирования, построения и функционирования распределенных систем. Уметь: – применять декомпозицию, проектирование взаимодействий, укрупнение и планирование вычислений при разработке параллельного алгоритма; – 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. Владеть: – способами преобразования циклов для ликвидации информационных зависимостей между итерациями; – навыками разработки, компиляции и отладки параллельных программ; – навыками освоения большого объема информации и решения задач распределенных систем.
ОПК-9 Способен осваивать методики использования программных средств для решения практических задач.	ОПК-9.4. Самостоятельно осваивает и применяет типовые и специализированные программные средства в соответствии с заданием.	Уметь: – самостоятельно осваивать и применять современные инструменты параллельного программирования.

1.2. Место дисциплины

Дисциплина включена в модуль «Программирование» ОПОП ВО, обязательная часть. Дисциплина осваивается на 4 курсе, в зимнюю сессию.

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

Таблица 2 – 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
	ЗФО
1 Общая трудоемкость дисциплины	108
2 Контактная работа обучающихся с преподавателем (по видам учебных занятий) (всего)	48
Аудиторная работа (всего):	12
в том числе:	
лекции	2
практические занятия, семинары	
практикумы	
лабораторные работы	10
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)	92
4 Промежуточная аттестация обучающегося и объём часов, выделенный на промежуточную аттестацию: – зачет	4

3. Учебно-тематический план и содержание дисциплины.

Таблица 3 - Учебно-тематический план заочной 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
			ЗФО			
			Аудиторн. занятия		СРС	
			лекц.	практ.		
Курс 4						
	1. Понятие параллельных и распределенных систем	**26**	**2**	**2**		Опрос №1
1	1.1 Определение и особенности распределённых систем					
2	1.2 Архитектура параллельных и					

№ недели п/п	Разделы и темы дисциплины по занятиям	Общая трудоёмкость *(всего час.)*	Трудоемкость занятий (час.)			Формы текущего контроля и промежуточной аттестации успеваемости
			ЗФО			
			Аудиторн. занятия		СРС	
			лекц.	практ.		
Курс 4						
	распределенных систем					
	2. Параллельное программирование	**26**		**4**		Контрольная работа. Индивидуальное задание №1 Опрос №2 Опрос №3 Опрос №4
3-5	2.1 Параллельные вычисления					
4-6	2.2 Технология параллельного программирования систем с общей памятью на OpenMP					
6	2.3 Использование графических процессоров		-			
7-9	2.4 Интерфейс передачи сообщений MPI		-			
	3. Модели распределенных систем	**26**		**2**		Контрольная работа. Индивидуальное задание №2 Индивидуальное задание №3
7-11	3.1 Модель распределенного исполнения					
9-12	3.2 Логическое время					
10-13	3.3 Синхронное и асинхронное исполнение					
11-14	3.4 Модели отказов					
12-15	3.5 Глобальное состояние					
	4. Распределенные системы	**26**		**2**		Контрольная работа. Индивидуальное задание №4 Индивидуальное задание №5 Индивидуальное задание №6
13	4.1 Коммуникационная подсистема					
14-16	4.2 Синхронизация					
15	4.3 Репликация и консистентность					
16	4.4 Безопасность					
17	4.5 Системы хранения данных					
18	4.6 Распределенные вычисления					
18	Промежуточная аттестация –зачет	**4**				4
	Всего:	**108**	**2**	**10**	**92**	**4**

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 4 - Балльно-рейтинговая оценка результатов учебной работы обучающихся по видам (БРС)

Учебная работа (виды)	Сумма баллов	Виды и результаты учебной работы	Оценка в аттестации	Баллы (17 недель)
Текущая учебная работа в семестре (Посещение занятий по расписанию и выполнение заданий)	**80**	Лекционные занятия (конспект)	посещение лекционного занятия	0 - 6
		Практические работы (отчет о выполнении практической работы)	посещение практического занятия и выполнение работы на 51-65% посещение занятия и существенный вклад на занятии в работу всей группы, самостоятельность и выполнение работы на 85,1-100%	12 – 24
		Контрольная работа. Индивидуальные задания	**За одно ИЗ :** **4 балла** (выполнено 51 - 65% заданий) **5 баллов** (выполнено 66 - 85% заданий) **7 баллов** (выполнено 86 - 100% заданий)	24 - 42
		Опросы (4 опроса)	**1 балла** (пороговое значение) **2 балла** (максимальное значение)	4 - 8
Итого по текущей работе в семестре				41 - 80
Промежуточная аттестация (зачет с оценкой)	20 (100% /баллов приведенной шкалы)	Тест.	**5 баллов** (пороговое значение) **10 баллов** (максимальное значение)	5 - 10
		Решение задачи.	**5 баллов** (пороговое значение) **10 баллов** (максимальное значение)	5 - 10
Итого по промежуточной аттестации (экзамену)				(51 – 100% по приведенной шкале) 10 – 20 б.
Суммарная оценка по дисциплине:		Сумма баллов текущей и промежуточной аттестации		51 – 100 б.

5 Учебно-методическое обеспечение дисциплины.

5.1 Учебная литература

Основная учебная литература

1. *Малявко, А. А.* Параллельное программирование на основе технологий openmp, cuda, opencl, mpi : учебное пособие для вузов / А. А. Малявко. — 3-е изд., испр. и доп. — Москва : Издательство Юрайт, 2023. — 135 с. — (Высшее образование). — ISBN 978-5-534-14116-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/514199

2. Бабичев, С. Л. Распределенные системы : учебное пособие для вузов / С. Л.

Дополнительная учебная литература

1. *Кудрина, Е. В.* Основы алгоритмизации и программирования на языке C# : учебное пособие для вузов / Е. В. Кудрина, М. В. Огнева. — Москва : Издательство Юрайт, 2023. — 322 с. — (Высшее образование). — ISBN 978-5-534-09796-2. — Текст :

электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/517285

2. *Бабичев, С. Л.* Распределенные системы : учебное пособие для вузов / С. Л. Бабичев, К. А. Коньков. — Москва : Издательство Юрайт, 2023. — 507 с. — (Высшее образование). — ISBN 978-5-534-11380-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/518274

5.2 Программное и информационное обеспечение освоения дисциплины.

5.2.1 Программное обеспечение

В обучении используются информационные технологии на базе компьютерных классов учебного корпуса №4 (пр.Металлургов 19):

- лекционные занятия ведутся с использованием презентаций и программного обеспечения мульти-медиа демонстраций на основе MicrosoftOffice 2010 (лицензия DreamSparkPremiumElectronicSoftwareDelivery (3 years);

Renewal по сублицензионному договору №Tr000083174 от 12.04.2016);

- практические занятия по дисциплине проводятся с использованием программного обеспечения, приведенного в таблице.

Таблица – Информационные технологии и программное обеспечение аудиторных занятий и самостоятельной работы

Программа / система	Сведения о праве использования (лицензия, договор, сроки использования).	№ комп. классов
Гарант	Гос.контракт №38 от 5.03.2010	501/4, 502/4, 508/4, 509/4,602 /4
ConsultantPlus 14	Договор об инфо поддержке 01.04.2007	501/4, 502/4, 508/4, 509/4,602/4
Программное обеспечение компьютеров: Операционные системы: Windows 7; Антивирусное ПО: EsetEndpointSecurity 5.0.		
Интерегрированные программые системы		
Microsoft Visual Studio 2010	Лиценизия DreamSpark Premium Electronic Software Delivery (3 years) Renewal по сублицензионному договору	501/4, 502/4, 508/4, 509/4,602/4
Офисное ПО		
Microsoft Office, Visio MS PowerPoint	ЛиценизияDreamSpark Premium Electronic Software Delivery (3 years) Renewal посублицензионномудоговору.	501/4, 502/4, 508/4, 509,602
Справочно-правовые и информационные системы		
Гарант	Гос.контракт №38 от 5.03.2010	501/4, 502/4, 508/4, 509/4,602/4
ConsultantPlus 14	Договор об инфо поддержке 01.04.2007	501/4, 502/4, 508/4, 509/4,602/4
Браузеры и дополнения		
IE 8	Бесплатно	501/4, 502/4, 508/4, 509/4,602/4
Firefox 14	Бесплатно	501/4, 502/4, 508/4, 509/4,602/4
Opera 12	Бесплатно	501/4, 502/4, 508/4, 509/4,602/4

Netbeans IDE 7.0.1 для Firefox	Бесплатно	501/4, 502/4, 508/4, 509/4,602/4
Специальное ПО для работы с компьютером лиц с ОВЗ		
NVDA	Бесплатно	501/4, 502/4, 508/4, 509/4,602/4
Экранная лупа, экранная клавиатура	В составе операционной системы	501/4, 502/4, 508/4, 509/4,602/4

5.3.2 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?

Контрольная работа

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

7

Таблица 5 - Примерные теоретические вопросы и практические задания к зачету

Разделы и темы	Примерные теоретические вопросы	Примерные практические задания
Разделы дисциплины		
1. Понятие параллельных и распределенных систем		
1.1 Определение и особенности распределённых систем	1. Особенности распределенных систем 2. Определение распределенной системы Отказоустойчивость в распределенных системах	
1.2 Архитектура параллельных и распределенных систем	3. Архитектура МКМД 4. Архитектура ОКОД 5. Сервисы распределенных систем 6. Архитектурные стили распределенных систем	
2. Параллельное программирование		
2.1 Параллельные вычисления	7. Блочное распараллеливание циклов 9. Развертка циклов	*Реализовать блочное распараллеливание циклов* *Реализовать преобразование циклов*

	10. Виды преобразования циклов	*расширением скаляра*
2.2 Технология параллельного программирования систем с общей памятью на OpenMP	11. Средства синхронизации потоков в OpenMD 12. Неявная барьерная синхронизация 13. Общие и локальные переменные потока	*Используя OpenMP, реализовать параллельный алгоритм*
2.3 Использование графических процессоров	14. Сетка, блок и поток 15. Ограничения на функции, исполняемые на GPU	
2.4 Интерфейс передачи сообщений MPI	16. Буферизированная передача сообщений 17. Коллективное взаимодействие ветвей программы 18. Исключающая редукция	*Реализовать параллельный ввод*
3. Модели распределенных систем		
3.1 Модель распределенного исполнения	19. Модель коммуникационного канала 20. Упорядочивание событий	
3.2 Логическое время	21. Логические часы 22. Скалярное время 23. Векторное время	*Реализовать алгоритм определения времени событий*
3.3 Синхронное и асинхронное исполнение	24. Асинхронное исполнение 25. Эмуляции	
3.4 Модели отказов	26. Отказы процессов 27. Иерархия моделей неисправности	*Построение модели отказов*
3.5 Глобальное состояние	28. Распределенная сборка мусора 29. Распределенное обнаружение завершения 30. Фиксация глобального состояния	
4. Распределенные системы		
4.1 Коммуникационная подсистема	31. Алгоритмы на графах 32. Удаленные вызовы	
4.2 Синхронизация	33. Алгоритмы синхронизации часов 34. Распределенные взаимоисключения	*Реализация алгоритма синхронизации*
4.3 Репликация и консистентность	35. Пассивная репликация 36. Активная репликация 37. Размещение и обновление реплик	
4.4 Безопасность	38. Модели контроля доступа 39. Дискретное управление доступом	
4.5 Системы хранения данных	40. Распределенные кластерные файловые системы 41. Пиринговые системы	
4.6 Распределенные вычисления	42. Модели распределённых вычислений. 43. Компоненты метасистемы	

Компетенции	
ОПК-2 Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;	*Кейс задание 1* Требуется реализовать параллельную реализацию алгоритма решения системы линейных алгебраических уравнений. - Определить технологии и алгоритмы для реализации эффективного алгоритма для системы с общей памятью - Определить и описать алгоритмы распараллеливания циклов для эффективной реализации алгоритма - Составить программный код параллельного алгоритма с использованием выбранных методов и технологий *Кейс задание 2* С использованием неблокирующих операций осуществить умножение квадратных матриц, используя различные виды распределения данных между процессами (выбрать из 64 вариантов использования топологий 5 наиболее эффективных, обосновать выбор и сравнить время выполнения).
ОПК-9 Способен осваивать методики использования программных средств для решения практических задач.	

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