

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Астраханский государственный университет имени В. Н. Татищева»
(Астраханский государственный университет им. В. Н. Татищева)

СОГЛАСОВАНО
Руководитель ОПОП

УТВЕРЖДАЮ
Зав. кафедрой ПМИ

_____ М.В. Коломина

_____ М.В. Коломина

«5» апреля 2024 г.

«5» апреля 2024 г.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
РАСПРЕДЕЛЕННОЕ ПРОГРАММИРОВАНИЕ**

Составитель	Корнеев Г.А., к.т.н., доцент ФИТиП, ИТМО
Согласовано с работодателями	Белов С.В., директор ООО «ТРАСТ ПОИНТ» Измайлов Г.А., генеральный директор ООО «Агент Плюс»
Направление подготовки / специальность	01.03.02 Прикладная математика и информатика
Направленность (профиль) ОПОП	Программирование и искусственный интеллект
Квалификация (степень)	бакалавр
Форма обучения	очная
Год приёма	2024
Курс	4
Семестр(ы)	8

1. ЦЕЛИ И ЗАДАЧИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

1.1. Целью освоения дисциплины (модуля) «Распределенное программирование» является изучение современных методов и средств обеспечения высокопроизводительных вычислений.

1.2. Задачи освоения дисциплины:

- сформировать навык проектирования и реализации параллельных и распределенных программных систем;
- углубление знаний в области программирования;
- развитие практических навыков в области распределенного программирования.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

2.1. Учебная дисциплина «Распределенное программирование» модуля «Элективные дисциплины» относится к части, формируемой участниками образовательных отношений и осваивается в 8 семестре.

2.2. Для изучения данной учебной дисциплины (модуля) необходимы следующие знания, умения, навыки, формируемые предшествующими учебными дисциплинами (модулями):

Язык программирования С++

Технологии программирования

Параллельное программирование

Знания: методы и технологии программирования, синтаксис и основные конструкции изучаемого языка программирования, базовые алгоритмы обработки данных, корректные постановки классических задач; аналитические и технологические решения в области программного обеспечения (системного, прикладного и инструментального) и компьютерной обработки информации.

Умения: разрабатывать алгоритмы, реализовывать алгоритмы на языке программирования, описывать основные структуры данных, реализовывать методы анализа и обработки данных, работать в средах программирования; диагностировать работоспособность вычислительной системы и устранять неполадки.

Навыки: владеть методами и технологиями разработки алгоритмов, описания структур данных и других базовых представлений данных, программирования на языке высокого уровня, навыками работы в некоторой среде программирования.

2.3. Последующие учебные дисциплины (модули) и (или) практики, для которых необходимы знания, умения, навыки, формируемые данной учебной дисциплиной (модулем):

- Написание выпускной квалификационной работы,
- Производственная практика.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ

Процесс освоения дисциплины направлен на формирование элементов следующих компетенций в соответствии с ФГОС ВО и ОПОП ВО по данному направлению подготовки / специальности:

б) профессиональных (ПК).

ПК-2. Способен осуществлять интеграцию программных модулей и компонент и проверку работоспособности кода программного обеспечения

ПК-4. Способен реализовывать программные средства

ПК-9. Способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения

ПК-10. Разработка и реализация архитектуры программного обеспечения

ПК-14. Способность определять эффективный способ решения прикладных задач с применением информационных технологий и программной инженерии, разрабатывать и внедрять соответствующие программные решения

ПК-15. Способность проектировать и реализовывать сложные программные системы и комплексы, в том числе высоконагруженные, распределенные и масштабируемые гетерогенные системы

Таблица 1 – Декомпозиция результатов обучения

Код компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине (модулю)		
		Знать (1)	Уметь (2)	Владеть (3)
ПК-2	ПК-2.1. Способен разрабатывать тестовые наборы данных. ПК-2.2. Способен проверять работоспособность программного обеспечения ПК-2.3. Способен осуществлять интеграцию программных модулей и компонентов и верификацию выпусков программного продукта	программные продукты, программные модули и компоненты и верификации выпусков программного продукта	выполнять проверку работоспособности программного обеспечения	навыками осуществления интеграции программных модулей
ПК-4	ПК-4.1. Способен анализировать качество кода. ПК-4.2. Способен проводить испытания программного средства и его компонентов ПК-4.3. Интеграция и внедрение разработанного программного обеспечения	инструменты для работы с программными требованиями	работать с интеграционными тестами	навыками написания интеграционных тестов
ПК-9	ПК-9.1. Владение базовыми навыками теории графов и алгоритмами на них ПК-9.2. Владение основами теории вычислимости и оценки сложности алгоритмов	базовые навыки теории графов и алгоритмы на них, основы теории вычислимости и оценки сложности алгоритмов	применять основы теории вычислимости и оценки сложности алгоритмов	Навыками разработки и применения алгоритмических и программных решений в области системного и прикладного программного обеспечения
ПК-10	ПК-10.1. Описание алгоритмов компонентов, включая методы и схемы. ПК-10.2. Описание технологии обработки данных для возможности их использования в программном средстве, включая вопросы параллельной обработки.	Алгоритмы компоненты, включая методы и схемы	Описать технологии обработки данных для возможности их использования в программном средстве	Навыками разработки и реализации архитектуры программного обеспечения
ПК-14	ПК-14.1. Способность проектировать и реализовывать программные решения с применением методов функционального, автоматного и эволюционного программирования ПК-14.2. Способность проектировать и реализовывать мобильные и web-приложения	методы функционального, автоматного и эволюционного программирования	проектировать и реализовывать мобильные и web-приложения	навыками определения эффективных способов решения прикладных задач с применением информационных технологий и программной инженерии, разработки и внедрения соответствующих программных решений
ПК-15	ПК-15.1. Способен проектировать и реализовывать параллельные и распределенные программные системы ПК-15.2. Способен применять алгоритмы и структуры данных для эффективной реализации высоконагруженных систем	программные системы	применять алгоритмы и структуры данных для эффективной реализации	навыками проектирования и реализации параллельных и распределенных программных систем

			высоконагруженных систем	
--	--	--	--------------------------	--

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Общая трудоемкость дисциплины в соответствии с учебным планом составляет 4 зачетные единицы (144 часа).

Трудоемкость отдельных видов учебной работы студентов очной формы обучения приведена в таблице 2.1.

Таблица 2.1. Трудоемкость отдельных видов учебной работы по формам обучения

Вид учебной и внеучебной работы	для очной формы обучения
Объем дисциплины в зачетных единицах	4
Объем дисциплины в академических часах	144
Контактная работа обучающихся с преподавателем (всего), в том числе (час.):	33
- занятия лекционного типа, в том числе:	11
- практическая подготовка (если предусмотрена)	0
- занятия семинарского типа (семинары, практические, лабораторные), в том числе:	22
- практическая подготовка (если предусмотрена)	0
- консультация (предэкзаменационная)	0
- промежуточная аттестация по дисциплине	0
Самостоятельная работа обучающихся (час.)	111
Форма промежуточной аттестации обучающегося (зачет/экзамен), семестр (ы)	Диф. зачет, 8 семестр

Содержание дисциплины, структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий и самостоятельной работы представлены в таблице 2.2.

Таблица 2.2. Структура и содержание дисциплины (модуля)

для очной формы обучения

Раздел, тема дисциплины (модуля)	Контактная работа, час.						КР / КП	СР, час.	Итого часов	Форма текущего контроля успеваемости, форма промежуточной аттестации [по семестрам]
	Л		ПЗ		ЛР					
	Л	в т.ч. ПП	ПЗ	в т.ч. ПП	ЛР	в т.ч. ПП				
Семестр 8										
Раздел 1. Основы параллельного программирования	3				5		27	35	Лабораторная работа №1, 2	
Раздел 2. Основные объекты в параллельном программировании	3				6		28	37	Лабораторная работа №3, 4, 5	
Раздел 3. Продвинутое алгоритмы	3				6		28	37	Лабораторная работа №6, 7	
Раздел 4. Многопоточные объекты	2				5		28	35	Лабораторная работа №8, 9	
Консультации										
Контроль промежуточной аттестации									Диф. зачет	
ИТОГО за семестр:	11				22		111	144		
Итого за весь период	11				22		111	144		

Примечание: Л – лекция; ПЗ – практическое занятие, семинар; ЛР – лабораторная работа; ПП – практическая подготовка; КР / КП – курсовая работа / курсовой проект; КПА – контроль промежуточной аттестации; КС – консультации; СР – самостоятельная работа

Таблица 3. Матрица соотношения разделов, тем учебной дисциплины (модуля) и формируемых компетенций

Раздел, тема дисциплины (модуля)	Кол-во часов	Код компетенции						Общее кол-во компетенций
		ПК-2	ПК-4	ПК-9	ПК-10	ПК-14	ПК-15	
Раздел 1. Основы параллельного программирования	35	+	+	+	+	+	+	6
Раздел 2. Основные объекты в параллельном программировании	37	+	+	+	+	+	+	6
Раздел 3. Продвинутое алгоритмы	37	+	+	+	+	+	+	6
Раздел 4. Многопоточные объекты	35	+	+	+	+	+	+	6
Итого	144							

Краткое содержание каждой темы дисциплины (модуля)

Основы параллельного программирования:

Основные понятия параллельного программирования. Алгоритмы взаимного исключения . Разделяемые регистры . Атомарный снимок состояния

Основные объекты в параллельном программировании:

Консенсус . Многопоточные (Thread-Safe) объекты . Базовые структуры данных в многопоточности . Модель памяти. Применение JMM

Продвинутое алгоритмы

Продвинутое алгоритмы. Реализация CASN. Анализ конфликтов. Анализ конфликтов стека

Многопоточные объекты

Создания многопоточных объектов. Многопоточные объекты Software. Transactional Memory. Транзакции

5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ПО ПРЕПОДАВАНИЮ И ОСВОЕНИЮ ДИСЦИПЛИНЫ

5.1. Указания для преподавателей по организации и проведению учебных занятий по дисциплине (модулю)

Лекционные занятия

Основной формой реализации теоретического обучения является лекция, которая представляет собой систематическое, последовательное изложение преподавателем-лектором учебного материала теоретического характера. Цель лекции – организация целенаправленной познавательной деятельности студентов по овладению программным материалом учебной дисциплины.

Порядок подготовки лекционного занятия включает в себя выполнение следующих этапов:

- изучение требований программы дисциплины;
- определение целей и задач лекции;
- разработка плана проведения лекции;
- подбор литературы (ознакомление с методической литературой, публикациями периодической печати по теме лекционного занятия);
- отбор необходимого и достаточного по содержанию учебного материала;
- определение методов, приемов и средств поддержания интереса, внимания, стимулирования творческого мышления студентов;
- написание конспекта лекции.

Лекция должна включать следующие разделы:

- формулировку темы лекции;
- указание основных изучаемых разделов или вопросов и предполагаемых затрат времени на их изложение;
- изложение вводной части;
- изложение основной части лекции;
- краткие выводы по каждому из вопросов;

- заключение;
- рекомендации литературных источников по излагаемым вопросам.

Лабораторные занятия

Лабораторное занятие – целенаправленная форма организации педагогического процесса, направленная на углубление научно-теоретических знаний и овладение определенными методами работы, в процессе которых вырабатываются умения и навыки выполнения тех или иных учебных действий в данной сфере науки. Они развивают научное мышление и речь, позволяют проверить знания студентов и выступают как средства оперативной обратной связи.

Правильно организованные лабораторные занятия ориентированы на решение следующих задач:

- обобщение, систематизация, углубление, закрепление полученных на лекциях и в процессе самостоятельной работы теоретических знаний по дисциплине (предмету);
- формирование практических умений и навыков, необходимых в будущей профессиональной деятельности, реализация единства интеллектуальной и практической деятельности;
- выработка при решении поставленных задач таких профессионально значимых качеств, как самостоятельность, ответственность, точность, творческая инициатива.

Состав заданий для лабораторного занятия должен быть спланирован с расчетом, чтобы за отведенное время они могли быть качественно выполнены большинством учащихся.

Лабораторные занятия должны так быть организованы, чтобы студенты ощущали нарастание сложности выполнения заданий, испытывали бы положительные эмоции от переживания собственного успеха в учении, поисками правильных и точных решений.

Самостоятельная работа

Самостоятельная работа – это вид учебной деятельности, которую студент совершает в установленное время и в установленном объеме индивидуально или в группе, без непосредственной помощи преподавателя (но при его контроле), руководствуясь сформированными ранее представлениями о порядке и правильности выполнения действий.

В учебном процессе образовательного учреждения выделяются два вида самостоятельной работы:

- аудиторная – выполняется на учебных занятиях, под непосредственным руководством преподавателя и по его заданию (выполнение самостоятельных работ; выполнение контрольных и практических работ; решение задач);
- внеаудиторная – выполняется по заданию преподавателя, но без его непосредственного участия (подготовка к аудиторным занятиям; изучение учебного материала, вынесенного на самостоятельную проработку; выполнение домашних заданий разнообразного характера; выполнение индивидуальных заданий, направленных на развитие у студентов самостоятельности и инициативы; подготовка к контрольной работе). Внеаудиторные самостоятельные работы представляют собой логическое продолжение аудиторных занятий, проводятся по заданию преподавателя, который инструктирует студентов и устанавливает сроки выполнения задания.

5.2. Указания для обучающихся по освоению дисциплины (модулю)

Лекция

- Лекция – основной вид обучения в вузе.
- В лекции излагаются основные положения теории, ее понятия и законы, приводятся факты, показывающие связь теории с практикой.
- Накануне лекции необходимо повторить содержание предыдущей лекции (а также теорию по изучаемой теме в школьных учебниках геометрии, если эта тема была представлена в них), а затем посмотреть тему очередной лекции по программе (по плану лекций).

- Полезно вести записи (конспекты) лекций: для непонятных вопросов оставлять место при работе над темой лекции с учебными пособиями.
- Записи лекций следует вести в отдельной тетради, оставляя место для дополнений во время самостоятельной работы.
- При конспектировании лекций выделяйте главы и разделы, параграфы, подчеркивайте основное.

Лабораторное занятие

- Лабораторное занятие – наиболее активный вид учебных занятий в вузе. Он предполагает самостоятельную работу над лекциями и учебными пособиями.
- К каждому лабораторному занятию нужно готовиться. Подготовку следует начинать с повторения теории (по записям лекций или по учебному пособию). После этого нужно решать задачи из предложенного домашнего задания.

Организация самостоятельной работы

Самостоятельность в учебной работе способствует развитию заинтересованности студента в изучаемом материале, вырабатывает у него умение и потребность самостоятельно получать знания, что весьма важно для специалиста с высшим образованием. Самостоятельная работа студентов представлена в следующих формах:

- работа с учебной литературой и конспектом лекций с целью подготовки к лабораторным занятиям, составление конспектов тем, выносимых на самостоятельную проработку;
- систематическое выполнение домашних работ.

Таблица 4. Содержание самостоятельной работы обучающихся для очной формы обучения

Номер раздела (темы)	Темы/вопросы, выносимые на самостоятельное изучение	Кол-во часов	Форма работы
Раздел 1	Основы параллельного программирования	27	Изучение теоретического материала. Подготовка к лабораторным работам.
Раздел 2	Основные объекты в параллельном программировании	28	Изучение теоретического материала. Подготовка к лабораторным работам.
Раздел 3	Продвинутые алгоритмы	28	Изучение теоретического материала. Подготовка к лабораторным работам.
Раздел 4	Многопоточные объекты	28	Изучение теоретического материала. Подготовка к лабораторным работам.

6. ОБРАЗОВАТЕЛЬНЫЕ И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

При реализации различных видов учебной работы по дисциплине «Распределенное программирование» могут использоваться электронное обучение и дистанционные образовательные технологии.

6.1. Образовательные технологии

Учебные занятия по дисциплине могут проводиться с применением информационно-телекоммуникационных сетей при опосредованном (на расстоянии) интерактивном взаимодействии обучающихся и преподавателя в режимах on-line или off-line в формах.

Таблица 5. Образовательные технологии, используемые при реализации учебных занятий

Раздел, тема дисциплины (модуля)	Форма учебного занятия		
	Лекция	Практическое занятие, семинар	Лабораторная работа
Раздел 1. Основы параллельного программирования	Обзорная лекция	Не предусмотрено	Выполнение лабораторной работы
Раздел 2. Основные объекты в параллельном программировании	Обзорная лекция	Не предусмотрено	Выполнение лабораторной работы

Раздел 3. Продвинутые алгоритмы	<i>Обзорная лекция</i>	<i>Не предусмотрено</i>	<i>Выполнение лабораторной работы</i>
Раздел 4. Многопоточные объекты	<i>Обзорная лекция</i>	<i>Не предусмотрено</i>	<i>Выполнение лабораторной работы</i>

6.2. Информационные технологии

При реализации различных видов учебной и внеучебной работы используются следующие информационные технологии:

- система управления обучением LMS Moodle;
- использование возможностей Интернета в учебном процессе (рассылка заданий, предоставление выполненных работ, ответы на вопросы, ознакомление обучающихся с оценками и т.д.);
- использование электронных учебников и различных сайтов (например, электронные библиотеки, журналы и т.д.) как источник информации;
- использование возможностей электронной почты;
- использование средств представления учебной информации (электронных учебных пособий, применение новых технологий для проведения занятий с использованием презентаций и т.д.);
- использование интерактивных средств взаимодействия участников образовательного процесса (технологии дистанционного или открытого обучения в глобальной сети);
- использование интегрированных образовательных сред, где главной составляющей являются не только применяемые технологии, но и содержательная часть, т.е. информационные ресурсы (доступ к мировым информационным ресурсам, на базе которых строится учебный процесс).

6.3. Современные профессиональные базы данных и информационные справочные системы

6.3.1. Программное обеспечение

Наименование программного обеспечения	Назначение
Adobe Reader	Программа для просмотра электронных документов
Платформа дистанционного обучения LMS Moodle	Виртуальная обучающая среда
Microsoft Office 2013, Microsoft Office Project 2013, Microsoft Office Visio 2013	Пакет офисных программ
7-zip	Архиватор
Microsoft Windows 7 Professional	Операционная система
Kaspersky Endpoint Security	Средство антивирусной защиты
Google Chrome	Браузер
OpenOffice	Пакет офисных программ

6.3.2. Современные профессиональные базы данных и информационные справочные системы

1. Электронная библиотека «Астраханский государственный университет» собственной генерации на платформе ЭБС «Электронный Читальный зал – BiblioTech». <https://biblio.asu.edu.ru>
2. Электронно-библиотечная система (ЭБС) ООО «Политехресурс» «Консультант студента». www.studentlibrary.ru.
3. Электронная библиотечная система издательства ЮРАЙТ, раздел «Легендарные книги». www.biblio-online.ru
4. Электронный каталог Научной библиотеки АГУ на базе MARK SQL НПО «Информ-систем». <https://library.asu.edu.ru>

7. ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕГО КОНТРОЛЯ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

7.1. Паспорт фонда оценочных средств

При проведении текущего контроля и промежуточной аттестации по дисциплине «Распределенное программирование» проверяется сформированность у обучающихся компетенций, указанных в разделе 3 настоящей программы. Этапность формирования данных компетенций в процессе освоения образовательной программы определяется последовательным освоением дисциплин и прохождением практик, а в процессе освоения дисциплины – последовательным достижением результатов освоения содержательно связанных между собой разделов, тем.

Таблица 6. Соответствие разделов, тем дисциплины, результатов обучения по дисциплине и оценочных средств

Контролируемые разделы, темы дисциплины (модуля)	Код контролируемой компетенции (компетенций)	Наименование оценочного средства
Основы параллельного программирования	ПК-2, ПК-4, ПК-9, ПК-10, ПК-14, ПК-15	лабораторные работы
Основные объекты в параллельном программировании	ПК-2, ПК-4, ПК-9, ПК-10, ПК-14, ПК-15	лабораторные работы
Продвинутое алгоритмы	ПК-2, ПК-4, ПК-9, ПК-10, ПК-14, ПК-15	лабораторные работы
Многопоточные объекты	ПК-2, ПК-4, ПК-9, ПК-10, ПК-14, ПК-15	лабораторные работы

7.2. Описание показателей и критериев оценивания компетенций, описание шкал оценивания

Таблица 7. Показатели оценивания результатов обучения в виде знаний

Шкала оценивания	Критерии оценивания
5 «отлично»	демонстрирует глубокое знание теоретического материала, умение обоснованно излагать свои мысли по обсуждаемым вопросам, способность полно, правильно и аргументированно отвечать на вопросы, приводить примеры
4 «хорошо»	демонстрирует знание теоретического материала, его последовательное изложение, способность приводить примеры, допускает единичные ошибки, исправляемые после замечания преподавателя
3 «удовлетворительно»	демонстрирует неполное, фрагментарное знание теоретического материала, требующее наводящих вопросов преподавателя, допускает существенные ошибки в его изложении, затрудняется в приведении примеров и формулировке выводов
2 «неудовлетворительно»	демонстрирует существенные пробелы в знании теоретического материала, не способен его изложить и ответить на наводящие вопросы преподавателя, не может привести примеры

Таблица 8. Показатели оценивания результатов обучения в виде умений и владений

Шкала оценивания	Критерии оценивания
5 «отлично»	демонстрирует способность применять знание теоретического материала при выполнении заданий, последовательно и правильно выполняет задания, умеет обоснованно излагать свои мысли и делать необходимые выводы
4 «хорошо»	демонстрирует способность применять знание теоретического материала при выполнении заданий, последовательно и правильно выполняет задания, умеет обоснованно излагать свои мысли и делать необходимые выводы, допускает единичные ошибки, исправляемые после замечания преподавателя
3 «удовлетворительно»	демонстрирует отдельные, несистематизированные навыки, испытывает затруднения и допускает ошибки при выполнении заданий, выполняет задание по подсказке преподавателя, затрудняется в формулировке выводов
2	не способен правильно выполнить задания

Шкала оценивания	Критерии оценивания
«неудовлетворительно»	

7.3. Контрольные задания и иные материалы, необходимые для оценки результатов обучения по дисциплине (модулю)

Контроль успеваемости по дисциплине осуществляется с помощью следующих оценочных средств:

Лабораторные работы

№ п/п	Номер раздела дисциплины	Наименование лабораторной работы
1	1	Алгоритмы взаимного исключения
2	1	Атомарный снимок состояния
3	2	Консенсус
4	2	Базовые структуры данных в многопоточности
5	2	Применение JMM
6	3	Реализация CASN
7	3	Анализ конфликтов стека
8	4	Многопоточные объекты
9	4	Транзакции

Примеры заданий к лабораторной работе «Алгоритмы взаимного исключения»

1. Студенты должны реализовать программу, реализующую алгоритмы взаимного исключения: Петерсона, Лампорта

Примеры заданий к лабораторной работе «Атомарный снимок состояния»

1. Студенты должны реализовать программу для построения атомарного снимка состояния N регистров

Примеры заданий к лабораторной работе «Консенсус»

1. Студенты должны реализовать визуализацию метода достижения консенсуса

Примеры заданий к лабораторной работе «Базовые структуры данных в многопоточности»

1. Студенты должны реализовать программу, реализующую одну из базовых структур данных в условиях многопоточности

Примеры заданий к лабораторной работе «Применение JMM»

1. Студенты должны реализовать программу с использованием JMM

Примеры заданий к лабораторной работе «Реализация CASN»

1. Студенты должны реализовать программу, реализующую CASN.

Примеры заданий к лабораторной работе «Анализ конфликтов стека»

1. Студенты должны реализовать программу для анализа конфликтов стека

Примеры заданий к лабораторной работе «Многопоточные объекты»

1. Студенты должны реализовать программу, использующую составные многопоточные объекты

Примеры заданий к лабораторной работе «Транзакции»

1. Студенты должны реализовать модель транзакций с блокировками

Вопросы, выносимые на дифференцированный зачет

1. Логические часы Лампорта и векторные часы, их свойства
2. Часы с прямой зависимостью (и их свойства) и матричные часы.
3. Взаимное исключение в распределенной системе. Централизованный алгоритм.
4. Взаимное исключение в распределенной системе. Алгоритм Лампорта,

5. Взаимное исключение в распределенной системе. Алгоритм Рикарта и Агравалы.
6. Взаимное исключение в распределенной системе. Алгоритм обедающих философов.
7. Взаимное исключение в распределенной системе. Алгоритм на основе токена.
8. Взаимное исключение в распределенной системе. Алгоритмы основе кворума (простое большинство, рушащиеся стены).
9. Согласованное глобальное состояние (согласованный срез). Алгоритм Чанди-Лампорта. Запоминание сообщений на стороне отправителя.
10. Согласованное глобальное состояние (согласованный срез). Алгоритм Чанди-Лампорта. Запоминание сообщений на стороне получателя.
11. Глобальные свойства. Стабильные и нестабильные предикаты. Слабый конъюнктивный предикат. Централизованный алгоритм.
12. Слабый конъюнктивный предикат. Распределенный алгоритм.
13. Диффундирующие вычисления. Останов. Алгоритм Дейксты и Шолтена.
14. Локально-стабильные предикаты, согласованные интервалы, барьерная синхронизация (3 алгоритма). Применение для определения взаимной блокировки.
15. Упорядочивание сообщений. Определения, иерархия порядков. Алгоритм для FIFO.
16. Упорядочивание сообщений. Определения, иерархия порядков. Алгоритм для причинносогласованного порядка.
17. Упорядочивание сообщений. Определения, иерархия порядков. Алгоритм для синхронного порядка.
18. Общий порядок (total order). Алгоритм Лампорта.
19. Общий порядок (total order). Алгоритм Скина.
20. Иерархия ошибок в распределенных системах. Отказ узла в асинхронной системе --- невозможность консенсуса (доказательство Фишера-Линча-Патерсона).
21. Консенсус в распределенных системах. Применение консенсуса: выбор лидера, terminating reliable broadcast.
22. Синхронные системы. Алгоритм для консенсуса в случае отказа заданного числа узлов.
23. Синхронные системы. Проблема византийских генералов. Алгоритм для $N \geq 4, f = 1$. Объяснить идею обобщения для $f > 1$.
24. Синхронные системы. Проблема византийских генералов. Невозможность решения при $N = 3, f = 1$.
25. Недетерминированные алгоритмы консенсуса. Алгоритм Бен-Ора.
26. Raos. Алгоритм, его свойства. Основные модификации.
27. Транзакции в распределенных системах. 2 Phase Locking, 2 Phase Commit

Таблица 9. Примеры оценочных средств с ключами правильных ответов

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
ПК-2. Способен осуществлять интеграцию программных модулей и компонент и проверку работоспособности кода программного обеспечения				
1.	Задание закрытого типа	<p><i>Выберите верный ответ.</i></p> <p>Какой из стандартов параллельного программирования предназначен для использования в системах с общей памятью?</p> <p>а. OpenMP б. MPI в. OpenMPI г. MPICH</p>	а	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
2.		<p><i>Выберите верные ответы.</i></p> <p>Какие из реализаций стандарта параллельного программирования предназначены для использования в системах с распределенной памятью?</p> <p>а. OpenMP б. MPI в. OpenMPI г. MPICH</p>	а, г	1-3
3.		<p><i>Выберите верные ответы.</i></p> <p>Какие языки программирования обладают встроенной поддержкой многопроцессности?</p> <p>а. С б. Java в. С# г. TBB</p>	б, в	1-3
4.		<p><i>Выберите верный ответ.</i></p> <p>К какой модели программирования относится модель, в которой все исполнители (потоки/процессоры) запускают одну программу, а разделение логики между ними основано на уникальных идентификаторах исполнителей?</p> <p>а. MPI б. Loop Parallelism в. SPMD г. Master/Worker</p>	в	1-3
5.		<p><i>Выберите верный ответ.</i></p> <p>Для чего предназначен шаблон Loop Parallelism?</p> <p>а. коммуникация внутри распределенной системы из вычислительных узлов, соединенных между собой по замкнутой цепи</p>	в	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		б. формирование очереди задач при конвейерных вычислениях в. распараллеливание циклов в программе путем распределения отдельных итераций между потоками г. балансировка нагрузки на процессоры при распределенных вычислениях		
6.	Задание открытого типа	Дайте определение понятию <i>логические часы Лампорта</i> .	Логическими часами Лампорта называется конкретная реализация такой функции C из множества событий распределенных систем (внутреннее событие, событие отправки сообщения и событие приема сообщения) в множество неотрицательных целых чисел. У каждого процесса заводится счётчик (исходное значение неважно и может даже отличаться в разных потоках) Правила: счётчик увеличивается перед каждым внутренним событием процесса; при отправке сообщения значение счётчика увеличивается и прикрепляется к сообщению; при получении сообщения значение счётчика процесса-получателя выставляется в максимум текущего и полученного значения и увеличивается на 1. Значением вышеупомянутой целочисленной функции на событии является значение счётчика, принадлежащей тому же потоку, что и событие. Стоит заметить, что логическое время не уникально в пределах всей системы, а уникально только в рамках своего потока.	8-10
7.		Дайте определение понятию <i>векторные часы</i>	Векторные часы — это функция $VC(e): E \rightarrow \mathbb{R}^n$ (из событий в вектор константного размера) такая, что для любых двух событий e и f следующие утверждения равносильны: <ul style="list-style-type: none"> • $e \rightarrow f$ (е произошло до f) • $VC(e) < VC(f)$ (все компоненты $VC(e)$ меньше всех компонент $VC(f)$) 	5-8

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
8.		Опишите алгоритм <i>векторных часов</i> .	<p>Алгоритм векторных часов можно построить из логических часов Лампорта, если попросить каждый процесс помнить счётчики всех процессов, а не только свой:</p> <p>каждый поток имеет целочисленный n-мерный вектор (n — количество потоков), проинициализированный нулями.</p> <p>в случае внутреннего события счётчик текущего процесса увеличивается на 1;</p> <p>перед отправкой сообщения внутренний счётчик, соответствующий текущему процессу, увеличивается на 1, и вектор целиком прикрепляется к сообщению;</p> <p>при получении сообщения счётчик текущего процесса увеличивается на 1, далее значения в текущем векторе выставляются в покомпонентный максимум от текущего и полученного.</p>	8-10
9.		В чем заключается отличие векторных часов от матричных?	<p>Матричные часы – это обобщение векторных часов.</p> <p>В отличие от векторных часов:</p> <p>Каждый процесс хранит вектор векторов (матрицу) целых чисел;</p> <p>При передаче сообщения передается вся имеющаяся матрица;</p> <p>В случае любого события, i-ый поток увеличивает на единицу свою компоненту матрицы, т.е $M[i][i]$;</p> <p>При приеме сообщения для всех строк, кроме строки, соответствующей номеру получившего потока ($myId$), берется покомпонентный максимум. А для $M[myId][j]$ берется максимум из $M[myId][j]$ и $W[srcId][j]$, где W - матрица, которая пришла, $srcId$ - номер потока, от которого пришла матрица W.</p> <p>Используя матричные часы, мы можем оценить нижнюю границу того, что знает другой поток.</p>	8-10
10.		Опишите <i>централизованный алгоритм</i> взаимного исключения.	<p>Централизованный алгоритм взаимного исключения:</p> <p>Центральный управляющий узел ведет очередь запросов на вход.</p> <p>Процесс, желающий войти в критический участок, посылает ему сообщение "запрос" и ждет, пока не получит сообщение "разрешение" (когда подходит его очередь).</p>	

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
			<p>Когда процесс выходит из критического участка, то посылает управляющему узлу сообщение "освобождение".</p> <p>Всего нужно послать три сообщения для каждого случая доступа к критическому ресурсу. Не масштабируется из-за необходимости иметь выделенного координатора.</p>	
ПК-4. Способен реализовывать программные средства				
11.	Задание закрытого типа	<p><i>Выберите верный ответ.</i></p> <p>Что отличает семафор от и канала?</p> <p>а. канал показывает взаимодействие, семафор передает данные без гонки данных</p> <p>б. семафор обеспечивает синхронизацию потоков, канал передает данные с отсутствием гонки данных</p> <p>в. канал обеспечивает взаимодействие, семафор передает данные с условием их взятия</p> <p>г. семафор обеспечивает взаимодействие, канал передает данные с условием их взятия</p>	г	1-3
12.		<p><i>Выберите верный ответ.</i></p> <p>Кто ожидает соединение при клиент-серверной системе?</p> <p>а. сервер</p> <p>б. семафор</p> <p>в. клиент</p> <p>г. канал</p>	а	1-3
13.		<p><i>Выберите верный ответ.</i></p> <p>Будущее в базах данных...</p> <p>а. сетевые базы данных, выполняющие удаленные процедуры</p> <p>б. распределенные базы данных</p>	г	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		<p>в. многопоточные системы управления базами данных</p> <p>г. базы данных ключ-значение, работающие в памяти</p>		
14.		<p><i>Выберите верный ответ.</i></p> <p>Пример распределенной системы</p> <p>а. база данных SQLite</p> <p>б. консольная программа для ввода-вывода элементов массива</p> <p>в. мобильное приложение, взаимодействующие с сервером в сети Интернет</p> <p>г. приложение для анализа данных на диске компьютера и шифрования</p>	в	1-3
15.		<p><i>Выберите верный ответ.</i></p> <p>Преимущества вызова удаленных сервисов для ERP систем?</p> <p>а. ERP системы не могут предоставить нужные возможности вызова удаленных сервисов</p> <p>б. обработка данных баз данных</p> <p>в. делегирование сложных функций, требующих математики, многопоточности или сложных действий, что сложно написать на базе модельного языка ERP системы</p> <p>г. необходимо написание семафоров и каналов для моделирования действий, спроектированных на UML диаграмме Activity</p>	в	1-3
16.	Задание открытого типа	Опишите алгоритм Лампорта.	Алгоритм Лампорта -- алгоритм взаимного исключения, работающий в случае, если все сообщения идут FIFO и использующий логические часы Лампорта (т.е. с каждым сообщением посылается ещё и временная метка).	8-10

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
			<p>Каждый поток поддерживает очередь запросов на вход в критическую секцию. Приоритет – <временная метка, номер потока> (т.е при равенстве временных меток берем тот поток, чей номер меньше, иначе возможна взаимная блокировка, если процессы будут общаться идеально симметрично). Когда поток хочет войти в критическую секцию, он: Добавляет свой запрос в свою очередь (т.е временную метку и номер потока) Посылает всем потокам запрос (req) Ждет от них ответа (ok) Получив все ответы, ждет, когда он станет первым в своей очереди, и входит в критическую секцию Выйдя из критической секции, удаляет свой запрос из своей очереди, посылает всем сообщение о том, что вышел (rel) Действия вне критической секции:</p> <p>При получении запроса от другого потока, запрос добавляется в очередь и запрашивающему потоку посылается ответ При получении release от другого потока, его запрос удаляется из очереди</p>	
17.		Опишите алгоритм Рикарта-Агравалы.	<p>Алгоритм Рикарта-Агравалы — алгоритм взаимного исключения, являющийся оптимизацией алгоритма Лампорта.</p> <p>Мы объединяем запросы rel и ok в один: вместо отправки ok сразу посылает только если не хотим входить в критическую секцию или сразу как только выйдем из секции.</p> <p>Когда процесс P_i хочет войти в критический участок, то рассылает всем сообщение req с текущей временной меткой.</p> <p>Когда процесс P_k получает от P_j запрос войти в критический участок: если он сам не посылал запрос, то посылает отклик;</p> <p>если он послал свой запрос, он сравнивает временные метки этих двух запросов и посылает отклик только если у его собственного запроса метка позже (на самом деле, при равенстве меток, нужно проверить, у кого</p>	8-10

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
			<p>больше номер, т.е. для отсутствия блокировок нужно ввести приоритет на потоках);</p> <p>в остальных случаях процесс P_k задерживает отправку отклика. Процесс может войти в критический участок только после получения откликов ото всех других узлов сети. После выхода из него он рассылает задержанные отклики на все ожидающие запросы.</p>	
18.		Сформулируйте задачу <i>обедающих философов</i> .	<p>За круглым столом сидят шесть философов (процессы). На столе шесть тарелок с рисом и шесть вилок.</p> <p>В каждый момент философ может либо хотеть есть (тогда ему необходимо две вилки), либо хотеть думать (для этого ему нужен только он сам). Философы не разговаривают между собой (только в начале, когда договариваются об алгоритме). Необходимо, чтобы все философы поели и не возникло драки за вилки.</p>	5-8
19.		Сформулируйте <i>обобщенную задачу обедающих философов</i> .	<p>В случае с философами у нас есть N процессов и граф конфликтов между ними из N ребер (одно ребро — один общий для двух процессов ресурс). Процессу для работы надо собрать все конфликтующие с другими потоками ресурсы.</p> <p>А взаимное исключение — это полный граф конфликтов, т.е есть вилка для каждой пары философов. Так что алгоритм для философов можно использовать и для задачи взаимного исключения.</p>	5-8
20.		Дайте определение понятию <i>кворум</i> .	<p>Кворум - это семейство Q подмножеств множества процессов 2^P причем:</p> <p>1. Q замкнуто относительно взятия надмножества, т.е.</p> <p>если $Q \ni A \subseteq B$, то $B \in Q$</p> <p>2. Любые два элемента (т.е подмножества относительно множества процессов) этого семейства имеют непустое пересечение.</p>	5-8
ПК-9. Способность к разработке и применению алгоритмических и программных решений в области системного и прикладного программного обеспечения				
21.	Задание закрытого типа	<p><i>Выберите верный ответ.</i></p> <p>Диаграмма линий жизни</p>	г	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		<p>а. показывает взаимодействия между процессам с помощью Акторов</p> <p>б. показывает взаимодействия между процессам путем показа всех состояний процесса</p> <p>в. показывает взаимодействия между процессам для всех вариантов взаимодействия</p> <p>г. показывает взаимодействия между процессам для одного из вариантов взаимодействия</p>		
22.		<p><i>Выберите верный ответ.</i></p> <p>Как можно создать поток в C#?</p> <p>а. с помощью System.Distributed.ParallelObject</p> <p>б. с помощью System.Distributed.ThreadObject</p> <p>в. с помощью System.Threading.Thread</p> <p>г. с помощью Thread.Distributed.Process</p>	г	1-3
23.		<p><i>Выберите верный ответ.</i></p> <p>Зачем необходимо ожидание рабочих потоков при решении параллельных задач?</p> <p>а. необходимо синхронизовать параллельные действия в распределенной системе для распределения задач между процессорами</p> <p>б. необходимо закрыть семафор, чтобы синхронизовать данные</p> <p>в. поскольку потоки могут остаться в операционной системе и забивать процессорное время</p> <p>г. поскольку необходимо дождаться промежуточных результатов и сделать редукцию результата</p>	г	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
24.		<p><i>Выберите верный ответ.</i></p> <p>Барьерная синхронизация – это...</p> <p>а. это средство в C# для создания потоков б. нужен для разделения заданий и сбора результатов в. нужен для создания потоков в многокомпонентной системе г. нужен для ожидания потоков в распределенной системе</p>	б	1-3
25.		<p><i>Выберите верный ответ.</i></p> <p>При программировании сервиса на C# используется...</p> <p>а. System.Threading.ThreadService б. специальный именованный семафор в. удаленные процедуры для работы с потоками г. атрибут класса [ServiceContract]</p>	б	1-3
26.	Задание открытого типа	<p>Дайте определение термину <i>кворум простого большинства</i>.</p>	<p>Кворум простого большинства — пример кворума, у которого любой элемент обладает мощностью строго больше половины.</p> <p>Если у нас есть 5 процессов P1..P5, то элементом кворума, как один из вариантов, будет подмножество {P1, P2, P3}. Таким образом, если P4 захочет попасть в критическую секцию, то ему будет достаточно получить подтверждение от {P1, P2, P3}.</p>	5-8
27.		<p>Дайте определение термину <i>кворум рушащейся стенки</i>.</p>	<p>Кворум рушащейся стенки (Crumbling walls quorum) — пример кворума с подмножествами размера порядка $O(\sqrt{n})$ вместо $O(n)$ как у кворума простого большинства.</p> <p>Структура: Процессы упорядочены в линии, по возможности равной длины. Минимальный элемент кворума — объединение всех процессов какой-то одной линии плюс по одному представителю из каждой линии ниже.</p>	5-8

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)																									
28.		Дайте определение понятию <i>срез</i> .	Мотивация: если у распределенной системы нет «глобального состояния», то как запомнить её состояние на диске, чтобы можно было продолжить работу после восстановления с диска? Пусть EE — множество событий с полным порядком (\ll) в рамках каждого процесса. Срез F — подмножество E такое, что $\forall f \in E, \forall g \in G : f \rightarrow g \Rightarrow f \in F$.	5-8																									
29.		Дайте определение понятию <i>согласованный срез</i> .	Мотивация: если у распределенной системы нет «глобального состояния», то как запомнить её состояние на диске, чтобы можно было продолжить работу после восстановления с диска? Пусть EE — множество событий с полным порядком (\ll) в рамках каждого процесса. Согласованный срез G — подмножество E такое, что $\forall f \in E, \forall g \in G : f \rightarrow g \Rightarrow f \in G$.	5-8																									
30.		Сформулируйте теорему <i>нижняя граница для срезов</i> .	Заметим, что если есть два согласованных среза G_1G_1 и G_2G_2 , то срез $G_1 \cap G_2 G_1 \cap G_2$ тоже согласован и, более того, $(G_1 \cap G_2) \leq G_1, G_2 (G_1 \cap G_2) \leq G_1, G_2$. Доказательство: рассмотрим, какие сообщения могут пересылаться между различными частями системы: <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>откуда/куда</th> <th>$G_1 \cap G_2$</th> <th>$G_1 \cap G_2$</th> <th>$G_1 \cap G_2$</th> <th>$G_1 \cap G_2$</th> </tr> </thead> <tbody> <tr> <td>$G_1 \cap G_2$</td> <td>+</td> <td>*</td> <td>*</td> <td>*</td> </tr> <tr> <td>$G_1 \cap G_2$</td> <td>-</td> <td>*</td> <td>-</td> <td>*</td> </tr> <tr> <td>$G_1 \cap G_2$</td> <td>-</td> <td>-</td> <td>*</td> <td>*</td> </tr> <tr> <td>$G_1 \cap G_2$</td> <td>-</td> <td>-</td> <td>-</td> <td>*</td> </tr> </tbody> </table> Заметим, что не существует сообщений, которые бы пересылались из $E \setminus (G_1 \cap G_2) E \setminus (G_1 \cap G_2)$ в $G_1 \cap G_2 G_1 \cap G_2$, что и требовалось.	откуда/куда	$G_1 \cap G_2$	+	*	*	*	$G_1 \cap G_2$	-	*	-	*	$G_1 \cap G_2$	-	-	*	*	$G_1 \cap G_2$	-	-	-	*	8-10				
откуда/куда	$G_1 \cap G_2$	$G_1 \cap G_2$	$G_1 \cap G_2$	$G_1 \cap G_2$																									
$G_1 \cap G_2$	+	*	*	*																									
$G_1 \cap G_2$	-	*	-	*																									
$G_1 \cap G_2$	-	-	*	*																									
$G_1 \cap G_2$	-	-	-	*																									
ПК-10. Разработка и реализация архитектуры программного обеспечения																													
31.	Задание закрытого типа	<i>Выберите верный ответ.</i> Проблема обедающих философов может быть решена... а. добавлением синхронизации с еще одной вилкой б. добавлением процессора-официанта в. добавлением семафора у первого философа г. добавлением философа и двух вилок	б	1-3																									
32.		<i>Выберите верный ответ.</i>	в	1-3																									

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		<p>Если нужно вызвать удаленную процедуру, возвращающую курсы валют на бирже со стороннего сервиса, нужно использовать технологию...</p> <p>а. удаленное ожидание с map-reduce б. семафор в. веб-сервис г. канал</p>		
33.		<p><i>Выберите верный ответ.</i></p> <p>МРI – это ...</p> <p>а. многопоточные интерфейсы инициализации общих данных б. UML диаграмма в. мульти-процессорные интерфейсы синхронизации сетевых сервисов WCF г. технология для программирования вычислительных систем для суперкомпьютеров</p>	г	1-3
34.		<p><i>Выберите верный ответ.</i></p> <p>Что такое удаленный вызов процедуры?</p> <p>а. процедура вызова потока для синхронизации удаленности б. вызов процедуры, работающей на другом узле с возвратом результата в. передача данных между 1с формами в многопоточном режиме г. вызов процедуры через некоторое время</p>	б	1-3
35.		<p><i>Выберите верный ответ.</i></p> <p>Проблема гонки данных это...</p> <p>а. проблема гонки обедающих философов за едой</p>	б	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		б. доступ к общим данным в программе из конкурирующих процессов/потоков в. доступ к общему семафору в ОС из конкурирующих процессов/потоков г. доступ к ядру процессора за данными в распределенной системе		
36.	Задание открытого типа	Дайте описание алгоритму Чанди-Ломпорта.	<p>Алгоритм Чанди-Лампорта получения согласованного среза с запоминанием на стороне получателя с учетом FIFO канала связи.</p> <p>В алгоритме мы ставим в соответствие каждому процессу белый или красный цвет. Изначально все процессы белые. Когда все процессы станут красными, мы получим согласованный срез.</p> <p>Есть инициатор (observer) с которого всё начинается. Он становится красным, запоминает свое состояние и посылает маркер всем другим процессам, с которыми у него есть канал. Только после этого он может снова обрабатывать полученные сообщения или же отправлять. Если процесс получил маркер и еще белый, то он обязан стать красным, записать свое локальное состояние и отправить остальным процессам маркеры. Иначе ничего делать не надо. Таким образом, если процесс только что стал красным (и записал своё состояние в согласованный срез), то все сообщения, которые он уже обработал, были посланы остальными процессами в моменты, которые учтены в согласованном срезе из-за FIFO.</p> <p>Будем называть сообщения красными, если они отправлены от красного процесса, аналогично для белого процесса. С технической точки зрения, при отправке сообщения или же маркера, будем добавлять цвет передаваемого сообщения.</p>	8-10
37.		Дайте определение понятию <i>глобальные свойства систем</i> .	<p>Глобальные свойства системы — это предикаты от согласованных срезов системы (состояния процессов плюс сообщения в пути), которые бывают двух видов:</p> <ul style="list-style-type: none"> • стабильные предикаты; • нестабильные предикаты. 	5-8

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
			Говорить "предикат сейчас верен" или "предикат сейчас неверен" некорректно, потому что нет "сейчас". Можно говорить "существует/не существует согласованный срез", но если у нас система ещё работает, то "не существует" мы можем заявить лишь на каком-то префиксе срезов системе.	
38.		Дайте определения стабильным и локальным предикатам.	Предикат PP является стабильным , если для любых согласованных срезов $G \leq HG \leq H$ из $P(G)P(G)$ следует $P(H)P(H)$. Предикат PP является локальным , если он зависит только от состояния одного конкретного процесса. Например, если у процесса есть состояние (переменная) x , то локальным предикатом может быть $(x < 0)$.	5-8
39.		Дайте определение понятию <i>слабый конъюнктивный предикат</i> .	Слабый конъюнктивный предикат (WCP) — предикат, имеющий вид конъюнкции локальных предикатов над состоянием каждого процесса. Слабый конъюнктивный предикат P истинен, если он истинен на хотя бы одном согласованном срезе	5-8
40.		Дайте определение <i>централизованному алгоритму для WCP</i> .	Централизованный алгоритм для WCP – алгоритм для поиска наименьшего (проще говоря, самого левого) согласованного среза в котором выполняется слабый конъюнктивный предикат. Если есть хотя бы один согласованный срез, в котором выполняется слабый конъюнктивный предикат, то такой срез существует и единственен (см. слабый конъюнктивный предикат). В централизованном алгоритме используются векторные часы. Срез задается набором векторных часов для всех процессов или просто вектором, в котором соответствующая компонента показывает время для соответствующего потока.	5-8
ПК-14. Способность определять эффективный способ решения прикладных задач с применением информационных технологий и программной инженерии, разрабатывать и внедрять соответствующие программные решения				
41.	Задание закрытого типа	<i>Выберите верный ответ.</i> Узел распределенной системы – это... а. ERP для сетевых потоков	г	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		<p>б. бизнес приложение для распределения задач между семафорами на мультипроцессорной системе</p> <p>в. связь потоков на процессоре</p> <p>г. устройство, на котором работает один из компонентов распределенной системы</p>		
42.		<p><i>Выберите верный ответ.</i></p> <p>Мар - reduce подход ...</p> <p>а. нужен для разделения заданий и сбора результатов</p> <p>б. нужен для ожидания потоков в распределенной системе</p> <p>в. это средство в C# для создания потоков</p> <p>г. нужен для создания потоков в многокомпонентной системе</p>	а	1-3
43.		<p><i>Выберите верный ответ.</i></p> <p>Что такое WCF?</p> <p>а. UML диаграмма коммуникаций</p> <p>б. технология работы с семафорами</p> <p>в. технология web сервисов в C#.net</p> <p>г. технология web сервисов в IC</p>	в	1-3
44.		<p><i>Выберите верный ответ.</i></p> <p>Зачем нужен семафор?</p> <p>а. для синхронизации дедлоков</p> <p>б. для решения задачи бесконечного ожидания</p> <p>в. для доступа к квантованию времени</p> <p>г. для синхронизации процессов и потоков</p>	г	1-3
45.		<p><i>Выберите верный ответ.</i></p> <p>Проблема «обедающих философов» – это...</p>	а	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		а. моделирует дедлок б. моделирует алгоритм выбора лидера в. моделирует семафор г. моделирует гонку данных		
46.	Задание открытого типа	Как можно уменьшит количество посылок о выполнении предиката?	Если в каком-то процессе выполнялся предикат, а потом он получил сообщение, то не надо ещё раз высылать сообщение координатору. Доказательство: если бы было решение, где новое состояние процесса после получения сообщения было границей искомого среза, то мы можем сдвинуть это состояние назад. Предикат всё ещё будет выполняться, а срез менее согласованным не станет, т.к. получения сообщений можно выкидывать безболезненно.	5-8
47.		Дайте определение <i>распределённому алгоритмы для WCP.</i>	Распределенный алгоритм для WCP – алгоритм для поиска наименьшего (проще говоря, самого левого) согласованного среза в котором выполняется слабый конъюнктивный предикат. В распределенном алгоритме используются векторные часы, как и в централизованном (они вообще похожи, рекомендуется сначала понять централизованный). В дополнение к каждому из NN процессов заведем еще NN координаторов (вместо одного на всех), где каждый процесс связан со своим координатором. Каждый процесс отправляет сообщения либо другим процессам, либо своему координатору (каждый раз, когда выполняется локальный предикат и увеличились векторные часы). Каждый координатор отправляет сообщения только другим координаторам.	8-10
48.		Дайте определение <i>диффундирующих вычислений.</i>	Диффундирующее вычисление — это вычисление, которое происходит следующим образом: <ul style="list-style-type: none"> • Каждый процесс бывает либо <i>активным</i>, либо <i>пассивным</i> • Если процесс получает сообщение, он становится активным • Только активный процесс может посылать сообщения • Активный процесс может в любой момент по своему желанию стать пассивным 	5-8

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
			<ul style="list-style-type: none"> Исходно есть ровно один активный процесс — инициатор 	
49.		В чем заключается основная идея алгоритма Дейкстры и Шолтена.	<p>Алгоритм Дейкстры и Шолтена решает задачу останова диффундирующего вычисления в распределённой системе.</p> <p>Основная идея: выстроить процессы в дерево: кто кого активизировал. Процесс добавляется в дерево (становится "красным"), когда становится активным, а удаляется (становится "зелёным"), когда он и все его потомки стали пассивными и там нет сообщений (т.е. поддерево закончило вычисления). Исходно дерево содержит только инициатора, а когда вычисление остановится, станет пустым (о чём узнает инициатор).</p> <p>Каждый процесс будет требовать подтверждения на каждое своё сообщение, чтобы можно было учесть сообщения в пути.</p>	5-8
50.		Запишите 4 гарантии порядка сообщений в иерархии порядков сообщений.	<p>В распределённых системах могут быть разные гарантии порядка доставки отправленных сообщений. Более того: иногда программисты могут неявно предполагать тот или иной порядок и очень удивляться, когда он нарушается.</p> <p>Мы рассматриваем четыре гарантии порядка сообщений, от более слабых к более сильным:</p> <ol style="list-style-type: none"> Асинхронная передачи: никаких гарантий, только <i>exactly-once delivery</i> FIFO (First In First Out order) Причинно-согласованный порядок (<i>causally consistent ordering</i>, от слова <i>cause</i>, а не <i>casual</i>) Синхронный порядок 	5-8
ПК-15. Способностью проектировать и реализовывать сложные программные системы и комплексы, в том числе высоконагруженные, распределенные и масштабируемые гетерогенные системы				
51.	Задание закрытого типа	<p><i>Выберите верный ответ.</i></p> <p>Что такое распределенная система?</p> <p>а. комплекс приложений для 1с, которые распределяют потоки по процессорам суперкомпьютера</p> <p>б. система финансового распределения задач в</p>	г	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		экономическом кластере между его узлами в. система распределения задачей исполнителям в бизнес информатике г. комплекс, представляющий собой многокомпонентные приложения на узлах, взаимодействующих по сети		
52.		<p><i>Выберите верный ответ.</i></p> <p>Если в системе 4 ядра, то сколько потоков необходимо создать для эффективной работы многопоточного приложения?</p> <p>а. 4 б. 2 в. 8 г. 100</p>	4	1-3
53.		<p><i>Выберите верный ответ.</i></p> <p>Многопоточное приложение на 2 ядрах для обработки 10 элементов массива работает дольше, чем однопоточное. Причина?</p> <p>а. необходимо использовать минимум 10 ядер б. необходимо использовать многопоточные приложения для обработки больших данных, на малых данных они работают неэффективно за счет времени на создания и синхронизацию потоков в. нужно перерабатывать алгоритм г. оно всегда будет работать дольше, так как занимает больше ресурсов процессора</p>	б	1-3
54.		<p><i>Выберите верный ответ.</i></p> <p>Подождать поток в C# можно...</p>	г	1-3

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
		а. с использованием метода StopThread б. с использованием метода WaitThread в. с использованием метода StopProcess г. с использованием метода Join		
55.		<i>Выберите верный ответ.</i> Что такое поток в программировании? а. абстракция, созданная для выполнения параллельных задач в рамках одного процесса. б. абстракция, созданная для выполнения барьерных задач в рамках одной распределенной системы. в. абстракция, созданная для выполнения распределенных задач в рамках одного процесса. г. абстракция, созданная для выполнения удаленных процедур в распределенной системе.	а	1-3
56.	Задание открытого типа	Дайте определение <i>синхронному порядку сообщений</i> .	В системе есть синхронный порядок сообщений, если всем сообщениям можно сопоставить время $T(m)$ (число) так, что верно: 1. Для любого сообщения m : $T(rcv(m)) = T(snd(m))$ (обозначается $T(m)$) 2. Для любых двух событий $e \rightarrow f$ верно $T(e) < T(f)$ (обратное может быть неверно), т.е. T является логическими часами.	5-8
57.		Дайте определение <i>барьерной синхронизации</i> .	Интервал $[G, H]$ ($G \subseteq H$) называется барьерно-синхронизированным, если для любых событий $e \in G$ и $f \notin H$ верно, что $e \rightarrow f$.	5-8
58.		Сформулируйте задачу взаимной блокировки.	Задача: есть процессы, они могут друг друга ждать (в этот момент они ничего не могут делать и не меняют своё внутреннее состояние). Мы хотим запустить алгоритм, который будет нам сообщать о появлении взаимных блокировок, когда процессы ждут друг	5-8

№ п/п	Тип задания	Формулировка задания	Правильный ответ	Время выполнения (в минутах)
			друга по циклу. Практическое применение — поиск ошибок в программах (динамические анализаторы распределённого кода, базы данных и SQL-запросы).	
59.		Опишите алгоритм для FIFO-порядка.	Алгоритм FIFO основан на нумерации сообщений: <ul style="list-style-type: none"> • Для каждой пары процессов нумеруем отправляемые сообщения натуральными числами. • Отправитель знает номер отправленного сообщения, получатель знает номер следующего ожидаемого сообщения. • К каждому сообщению прикрепляется номер. • Если получатель получил "неожиданное" сообщение, то он складывает его в очередь и ждёт. Обрабатывает только когда подойдёт нужный номер. 	5-8
60.		Дайте определение термину <i>кворум простого большинства</i> .	Кворум простого большинства — пример кворума, у которого любой элемент обладает мощностью строго больше половины. Если у нас есть 5 процессов P1..P5, то элементом кворума, как один из вариантов, будет подмножество {P1, P2, P3}. Таким образом, если P4 захочет попасть в критическую секцию, то ему будет достаточно получить подтверждение от {P1, P2, P3}.	5-8
61.	Задание комбинированного типа	<i>Верно ли утверждение:</i> Многопоточное приложение на 2 ядрах для обработки 10 элементов массива работает дольше, чем однопоточное. Причина в том, что необходимо использовать минимум 10 ядер. Ответ обоснуйте.	Утверждение неверно, причина может заключаться в том, что необходимо использовать многопоточные приложения для обработки больших данных, на малых данных они работают неэффективно за счет времени на создания и синхронизацию потоков.	3-5

7.4. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине (модулю)

Таблица 10. Технологическая карта рейтинговых баллов по дисциплине

№ п/п	Контролируемые мероприятия	Количество мероприятий / баллы	Максимальное количество баллов	Срок представления
Основной блок				
1.	<i>Лабораторные работы</i>	9/10	90	
Всего			90	-
Блок бонусов				

№ п/п	Контролируемые мероприятия	Количество мероприятий / баллы	Максимальное количество баллов	Срок представления
2.	<i>Посещение занятий</i>		5	
3.	<i>Своевременное выполнение всех заданий</i>		5	
Всего			10	-
ИТОГО			100	-

Таблица 10. Шкала перевода рейтинговых баллов в итоговую оценку за семестр по дисциплине

Сумма баллов	Оценка по 4-балльной шкале	
90–100	5 (отлично)	Зачтено
85–89	4 (хорошо)	
75–84		
70–74		
65–69	3 (удовлетворительно)	
60–64		
Ниже 60	2 (неудовлетворительно)	Не зачтено

При реализации дисциплины в зависимости от уровня подготовленности обучающихся могут быть использованы иные формы, методы контроля и оценочные средства, исходя из конкретной ситуации.

8. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

8.1. Основная литература

1. Hopcroft J. E., Motwani R., Ullman J. D. Introduction to Automata Theory, Languages, and Computation (3rd Edition). — Addison-Wesley, Boston, MA, USA, 2006. — 750 с.
2. Шень А. Программирование: теоремы и задачи. — М.: МЦНМО, 2014. — 296 с.
3. Шень А., Верещагин Н. Языки и исчисления. — М.: МЦНМО, 2012. — 240 с.
4. Верещагин, Н. К. Колмогоровская сложность и алгоритмическая случайность [Электронный ресурс] / Н. К. Верещагин, В. А. Успенский, А. Шень. — Электрон. дан. — СПб: Лань, 2013. — 575 с. — Режим доступа: <https://e.lanbook.com/book/56395> — Загл. с экрана.

8.2. Дополнительная литература

1. Кривцова, И. Е. Основы дискретной математики. Часть 1. Учебное пособие [Электронный ресурс] / И. Е. Кривцова, И. С. Лебедев, А. В. Настека. — Электрон. дан. — СПб: ИТМО, 2016. — 92 с. — Режим доступа: http://books.ifmo.ru/book/1869/osnovy_diskretnoy_matematiki_chast_1_uchebnoe_posobie.htm — Загл. с экрана.
2. Ганичева, А. В. Дискретная математика : учебное пособие для вузов / А. В. Ганичева, А. В. Ганичев. — 2-е изд., стер. — Санкт-Петербург : Лань, 2024. — 160 с. — ISBN 978-5-507-49204-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/382370>
3. Шевелев, Ю. П. Дискретная математика : учебное пособие для вузов / Ю. П. Шевелев. — 5-е изд., стер. — Санкт-Петербург : Лань, 2024. — 592 с. — ISBN 978-5-507-49681-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/399194>
4. Вики-конспекты. — http://neerc.ifmo.ru/wiki/index.php?title=Заглавная_страница

8.3. Интернет-ресурсы, необходимые для освоения дисциплины (модуля)

Электронный каталог Научной библиотеки АГУ на базе MARK SQL НПО «Информ-систем»: <https://library.asu.edu.ru>

1. Корпоративный проект Ассоциации региональных библиотечных консорциумов (АРБИКОН) «Межрегиональная аналитическая роспись статей» (МАРС): <http://mars.arbicon.ru>

2. Единое окно доступа к образовательным ресурсам <http://window.edu.ru>

9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Для проведения лекционных занятий используется аудитория, оборудованная современной презентационной техникой (проектор, экран, ноутбук).

Для выполнения лабораторных работ используются компьютерные классы с установленным в них необходимым программным обеспечением.

10. ОСОБЕННОСТИ РЕАЛИЗАЦИИ ДИСЦИПЛИНЫ (МОДУЛЯ) ПРИ ОБУЧЕНИИ ИНВАЛИДОВ И ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

Рабочая программа дисциплины (модуля) при необходимости может быть адаптирована для обучения (в том числе с применением дистанционных образовательных технологий) лиц с ограниченными возможностями здоровья, инвалидов. Для этого требуется заявление обучающихся, являющихся лицами с ограниченными возможностями здоровья, инвалидами, или их законных представителей и рекомендации психолого-медико-педагогической комиссии. При обучении лиц с ограниченными возможностями здоровья учитываются их индивидуальные психофизические особенности. Обучение инвалидов осуществляется также в соответствии с индивидуальной программой реабилитации инвалида (при наличии).

Для лиц с нарушением слуха возможно предоставление учебной информации в визуальной форме (краткий конспект лекций; тексты заданий, напечатанные увеличенным шрифтом), на аудиторных занятиях допускается присутствие ассистента, а также сурдопереводчиков и тифлосурдопереводчиков. Текущий контроль успеваемости осуществляется в письменной форме: обучающийся письменно отвечает на вопросы, письменно выполняет практические задания. Доклад (реферат) также может быть представлен в письменной форме, при этом требования к содержанию остаются теми же, а требования к качеству изложения материала (понятность, качество речи, взаимодействие с аудиторией и т. д.) заменяются на соответствующие требования, предъявляемые к письменным работам (качество оформления текста и списка литературы, грамотность, наличие иллюстрационных материалов и т. д.). Промежуточная аттестация для лиц с нарушениями слуха проводится в письменной форме, при этом используются общие критерии оценивания. При необходимости время подготовки к ответу может быть увеличено.

Для лиц с нарушением зрения допускается аудиальное предоставление информации, а также использование на аудиторных занятиях звукозаписывающих устройств (диктофонов и т. д.). Допускается присутствие на занятиях ассистента (помощника), оказывающего обучающимся необходимую техническую помощь. Текущий контроль успеваемости осуществляется в устной форме. При проведении промежуточной аттестации для лиц с нарушением зрения тестирование может быть заменено на устное собеседование по вопросам.

Для лиц с ограниченными возможностями здоровья, имеющих нарушения опорно-двигательного аппарата, на аудиторных занятиях, а также при проведении процедур текущего контроля успеваемости и промежуточной аттестации могут быть предоставлены необходимые технические средства (персональный компьютер, ноутбук или другой гаджет); допускается присутствие ассистента (ассистентов), оказывающего обучающимся необходимую техническую помощь (занять рабочее место, передвигаться по аудитории, прочитать задание, оформить ответ, общаться с преподавателем).