

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
Федеральное государственное автономное образовательное учреждение высшего образования  
"Казанский (Приволжский) федеральный университет"  
Институт информационных технологий и интеллектуальных систем



*подписано электронно-цифровой подписью*

## **Программа дисциплины**

Распределенные вычисления и приложения

Направление подготовки: 09.03.04 - Программная инженерия  
Профиль подготовки: Современная разработка программного обеспечения  
Квалификация выпускника: бакалавр  
Форма обучения: очное  
Язык обучения: русский  
Год начала обучения по образовательной программе: 2025

## Содержание

1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО
2. Место дисциплины (модуля) в структуре ОПОП ВО
3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся
4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий
  - 4.1. Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)
  - 4.2. Содержание дисциплины (модуля)
5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)
6. Фонд оценочных средств по дисциплине (модулю)
7. Перечень литературы, необходимой для освоения дисциплины (модуля)
8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)
9. Методические указания для обучающихся по освоению дисциплины (модуля)
10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)
11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)
12. Средства адаптации преподавания дисциплины (модуля) к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья
13. Приложение №1. Фонд оценочных средств
14. Приложение №2. Перечень литературы, необходимой для освоения дисциплины (модуля)
15. Приложение №3. Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем

Программу дисциплины разработал(а)(и): старший преподаватель, б/с Насибуллина Э.Р. (Кафедра программной инженерии, Институт информационных технологий и интеллектуальных систем), ERStepanova@kpfu.ru ; доцент, к.н. Шемахин А.Ю. (Кафедра радиофизики, Высшая школа киберфизических систем и прикладной электроники), ashemakhin@yandex.ru

# **1. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с планируемыми результатами освоения ОПОП ВО**

Обучающийся, освоивший дисциплину (модуль), должен обладать следующими компетенциями:

<b>Шифр компетенции</b>	<b>Расшифровка приобретаемой компетенции</b>
ПК-7	Владение навыками использования операционных систем, сетевых технологий, средств разработки программного интерфейса, применения языков и методов формальных спецификаций, систем управления базами данных

Обучающийся, освоивший дисциплину (модуль):

Должен знать:

Основные концепции и архитектуры распределенных систем

Модели распределенных вычислений и их применение

Основы сетевого взаимодействия и протоколы распределенных систем

Архитектуры и типы распределенных приложений (клиент-сервер, P2P, SOA, микросервисы)

Механизмы обеспечения надежности распределенных систем (репликация, отказоустойчивость)

Язык расширяемой разметки XML и его применение в распределенных системах

Методы формальной спецификации для описания распределенных систем

Системы управления базами данных в распределенной среде

Безопасность и масштабируемость распределенных приложений

Современные платформы и инструменты для разработки распределенных приложений

Должен уметь:

Проектировать архитектуру распределенных систем

Разрабатывать распределенные приложения с использованием различных парадигм

Работать с сетевыми протоколами и API для распределенного взаимодействия

Применять формальные методы для спецификации и верификации

Использовать операционные системы для развертывания распределенных систем

Работать с базами данных в распределенной архитектуре

Выбирать подходящие инструменты для разработки распределенных приложений

Анализировать и оптимизировать производительность распределенных систем

Должен владеть:

Практическими навыками разработки распределенных приложений

Опытом работы с операционными системами и сетевыми технологиями

Знанием методов формальной спецификации и верификации

Навыками работы с системами управления базами данных в распределенной среде

Опытом использования инструментов для разработки распределенных приложений

Способностью анализировать требования и проектировать решения

Навыками тестирования и отладки распределенных систем

## 2. Место дисциплины (модуля) в структуре ОПОП ВО

Данная дисциплина (модуль) включена в раздел "Б1.В.ДВ.05.10 Дисциплины (модули)" основной профессиональной образовательной программы 09.03.04 "Программная инженерия (Современная разработка программного обеспечения)" и относится к дисциплинам по выбору части ОПОП ВО, формируемой участниками образовательных отношений.

Осваивается на 3 курсе в 6 семестре.

## 3. Объем дисциплины (модуля) в зачетных единицах с указанием количества часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся

Общая трудоемкость дисциплины составляет 5 зачетных(ые) единиц(ы) на 180 часа(ов).

Контактная работа - 72 часа(ов), в том числе лекции - 0 часа(ов), практические занятия - 72 часа(ов), лабораторные работы - 0 часа(ов), контроль самостоятельной работы - 0 часа(ов).

Самостоятельная работа - 72 часа(ов).

Контроль (зачёт / экзамен) - 36 часа(ов).

Форма промежуточного контроля дисциплины: экзамен в 6 семестре.

## 4. Содержание дисциплины (модуля), структурированное по темам (разделам) с указанием отведенного на них количества академических часов и видов учебных занятий

### 4.1 Структура и тематический план контактной и самостоятельной работы по дисциплине (модулю)

N	Разделы дисциплины / модуля	Се-местр	Виды и часы контактной работы, их трудоемкость (в часах)						Само-стоя-тель-ная ра-бота
			Лекции, всего	Лекции в эл. форме	Практи-ческие занятия, всего	Практи-ческие в эл. форме	Лабора-торные работы, всего	Лабора-торные в эл. форме	
1.	Тема 1. Основные концепции распределенных систем	6	0	0	8	0	0	0	8
2.	Тема 2. Архитектуры и модели распределенных вычислений	6	0	0	8	0	0	0	8
3.	Тема 3. Сетевые технологии и протоколы распределенных систем	6	0	0	8	0	0	0	8
4.	Тема 4. Архитектура клиент-сервер и web-сервисы	6	0	0	8	0	0	0	8
5.	Тема 5. Peer-to-Peer архитектура и децентрализованные системы	6	0	0	8	0	0	0	8
6.	Тема 6. Сервис-ориентированная архитектура и микросервисы	6	0	0	8	0	0	0	8

N	Разделы дисциплины / модуля	Семестр	Виды и часы контактной работы, их трудоемкость (в часах)						Самостоятельная работа
			Лекции, всего	Лекции в эл. форме	Практические занятия, всего	Практические в эл. форме	Лабораторные работы, всего	Лабораторные в эл. форме	
7.	Тема 7. Распределенные базы данных и управление данными	6	0	0	8	0	0	0	8
8.	Тема 8. Надежность и отказоустойчивость распределенных систем	6	0	0	8	0	0	0	8
9.	Тема 9. XML и методы формальной спецификации	6	0	0	8	0	0	0	8
	Итого		0	0	72	0	0	0	72

## 4.2 Содержание дисциплины (модуля)

### Тема 1. Основные концепции распределенных систем

Определение распределенной системы как совокупности независимых компьютеров, связанных сетью и управляемых общей программой. История развития: от централизованных систем через локальные сети к облачным системам и IoT. Классификация по топологии, типам узлов, синхронизации и моделям отказов. Характеристики: масштабируемость, надежность, производительность, прозрачность, гетерогенность. Парадигмы разработки: синхронные (глобальные часы, предсказуемые задержки) и асинхронные (нет глобальных часов, непредсказуемые задержки). Проблемы синхронизации и согласованности в распределенной среде. Основные вызовы: координация, отказоустойчивость, управление состоянием. Примеры современных распределенных систем: Интернет, облачные вычисления (AWS, Azure), социальные сети, IoT системы.

### Тема 2. Архитектуры и модели распределенных вычислений

Модели обработки данных: batch processing (MapReduce, Hadoop), stream processing (Apache Flink, Kafka Streams), graph processing (GraphX). Облачные модели обслуживания: IaaS (полный контроль над инфраструктурой), PaaS (разработка без управления инфраструктурой), SaaS (полностью управляемые приложения). Apache Spark как универсальная платформа обработки данных с кэшированием в памяти. Lambda архитектура: комбинация batch и stream слоев для полной аналитики. Архитектурные паттерны: Event sourcing (сохранение всех изменений как событий), CQRS (разделение операций чтения и записи), Saga pattern (управление распределенными транзакциями). Микробатчинг для приблизительной обработки в реальном времени.

### Тема 3. Сетевые технологии и протоколы распределенных систем

Протоколы сетевого уровня: TCP (надежная, упорядоченная доставка) и UDP (быстрая, ненадежная доставка). Протоколы прикладного уровня: HTTP/HTTPS (основа web-сервисов), gRPC (высокопроизводительный, использует Protocol Buffers), WebRTC (для peer-to-peer коммуникации). RPC (Remote Procedure Call) и RMI парадигмы для вызова функций на удаленных компьютерах. Асинхронная коммуникация через очереди сообщений и pub-sub системы. Message brokers: RabbitMQ (надежная доставка), Apache Kafka (высокопроизводительный журнал событий), AWS SQS. WebSockets для реал-тайм полнодуплексной коммуникации. Балансировка нагрузки и маршрутизация в распределенных сетях. Выбор протокола в зависимости от требований к надежности, скорости, простоте.

### Тема 4. Архитектура клиент-сервер и web-сервисы

Классическая двухуровневая архитектура клиент-сервер и трехуровневые многоуровневые архитектуры с разделением ответственности. Web-сервисы: SOAP (сложный, XML-based) и REST (простой, использует HTTP методы). RESTful принципы: resources как URIs, методы HTTP (GET, POST, PUT, DELETE), status codes для индикации результата. JSON как основной формат обмена данными вместо XML. API Gateway как единая точка входа, балансировка нагрузки, rate limiting, аутентификация. Аутентификация и авторизация: OAuth 2.0 для делегированной авторизации, JWT токены, API ключи. GraphQL как альтернатива REST для гибких запросов данных. API versioning и backward compatibility. Практическое проектирование REST API с использованием инструментов (Swagger, OpenAPI).

### Тема 5. Peer-to-Peer архитектура и децентрализованные системы

P2P архитектуры: неструктурированные (Gnutella), структурированные (Chord, Kademlia с DHT), гибридные с супер-узлами. BitTorrent как пример успешной P2P системы для обмена файлами. IPFS (децентрализованная файловая система) использует DHT и содержание-адресуемое хранилище. Distributed Hash Tables (DHT) для эффективного хранения и поиска данных на множестве узлов. Блокчейн как распределенный реестр с неизменяемостью через криптографию. Консенсус алгоритмы: Proof of Work (PoW), Proof of Stake (PoS). Смарт-контракты как программы на блокчейне. Децентрализованные приложения (DApps) без центрального органа управления.

### Тема 6. Сервис-ориентированная архитектура и микросервисы

Service-Oriented Architecture (SOA): разделение приложения на независимые услуги, ESB (Enterprise Service Bus). Микросервисная архитектура как эволюция SOA: независимость развертывания, слабая связанность, организация вокруг бизнес-возможностей. Контейнеризация с Docker для упаковки микросервисов со всеми зависимостями. Kubernetes для автоматического развертывания, масштабирования и управления контейнерами. Pods (наименьшие развертываемые единицы), Services (обнаружение и балансировка нагрузки), Ingress. Service Mesh (Istio) для управления микросервисами на уровне сети. Паттерны: Service Registry, API Gateway, Circuit Breaker, Saga для распределенных транзакций, Event Sourcing, CQRS. Challenges микросервисов: сложность, distributed tracing, monitoring, consistency.

#### **Тема 7. Распределенные базы данных и управление данными**

Проблемы распределенных баз данных: репликация (обновление копий), партиционирование (разделение данных), консистентность, доступность. ACID свойства vs BASE модель для распределенных систем. Шардирование (горизонтальное разделение) данных между узлами на основе хеша или диапазонов. NoSQL БД: Cassandra (высокая доступность, eventual consistency), MongoDB (документ-ориентированная), CouchDB. Google Spanner (сильная консистентность), DynamoDB (управляемый сервис). Consensus алгоритмы Paxos и Raft для обеспечения консистентности. CAP теорема: выбор двух из трех свойств (Consistency, Availability, Partition tolerance).

#### **Тема 8. Надежность и отказоустойчивость распределенных систем**

Модели отказов: fail-stop (узел останавливается), crash (может потерять состояние), omission (пропуск сообщений), Byzantine (произвольное поведение). Обнаружение отказов: heartbeat сообщения, timeout. Failover автоматическое переключение на резервный узел, graceful degradation. Репликация: master-slave (один лидер), master-master (несколько лидеров), quorum-based (большинство голосов). Мониторинг: метрики (CPU, память, сетевой трафик), логирование (ELK stack), трассировка (Jaeger, Zipkin), алертинг. Prometheus для сбора метрик, Grafana для визуализации. Отечественные решения для надежных систем. Disaster recovery и backup стратегии. Post-mortems анализ отказов для улучшения надежности.

#### **Тема 9. XML и методы формальной спецификации**

XML для структурированного представления данных с использованием тегов и атрибутов. XML Schema для определения структуры и типов данных. XPath язык для навигации по XML, XQuery для запросов, XSLT для трансформации. Применение: конфигурационные файлы, обмен данными, SOAP web-сервисы. Формальные методы спецификации: Z notation (математическая нотация), Alloy (язык моделирования), Temporal Logic. Model Checking: автоматическая верификация свойств системы. Инструменты: SPIN (model checker), TLA+ (Lamport's Temporal Logic of Actions). Верификация консенсус алгоритмов (Paxos, Raft), обнаружение race conditions и deadlocks. Практическое применение формальных методов в критичных распределенных системах.

### **5. Перечень учебно-методического обеспечения для самостоятельной работы обучающихся по дисциплине (модулю)**

Самостоятельная работа обучающихся выполняется по заданию и при методическом руководстве преподавателя, но без его непосредственного участия. Самостоятельная работа подразделяется на самостоятельную работу на аудиторных занятиях и на внеаудиторную самостоятельную работу. Самостоятельная работа обучающихся включает как полностью самостоятельное освоение отдельных тем (разделов) дисциплины, так и проработку тем (разделов), осваиваемых во время аудиторной работы. Во время самостоятельной работы обучающиеся читают и конспектируют учебную, научную и справочную литературу, выполняют задания, направленные на закрепление знаний и отработку умений и навыков, готовятся к текущему и промежуточному контролю по дисциплине.

Организация самостоятельной работы обучающихся регламентируется нормативными документами, учебно-методической литературой и электронными образовательными ресурсами, включая:

Порядок организации и осуществления образовательной деятельности по образовательным программам высшего образования - программам бакалавриата, программам специалитета, программам магистратуры (утвержден приказом Министерства науки и высшего образования Российской Федерации от 6 апреля 2021 года №245)

Письмо Министерства образования Российской Федерации №14-55-99бин/15 от 27 ноября 2002 г. "Об активизации самостоятельной работы студентов высших учебных заведений"

Устав федерального государственного автономного образовательного учреждения "Казанский (Приволжский) федеральный университет"

Правила внутреннего распорядка федерального государственного автономного образовательного учреждения высшего профессионального образования "Казанский (Приволжский) федеральный университет"

Локальные нормативные акты Казанского (Приволжского) федерального университета

### **6. Фонд оценочных средств по дисциплине (модулю)**

Фонд оценочных средств по дисциплине (модулю) включает оценочные материалы, направленные на проверку освоения компетенций, в том числе знаний, умений и навыков. Фонд оценочных средств включает оценочные средства текущего контроля и оценочные средства промежуточной аттестации.

В фонде оценочных средств содержится следующая информация:



- соответствие компетенций планируемым результатам обучения по дисциплине (модулю);
- критерии оценивания сформированности компетенций;
- механизм формирования оценки по дисциплине (модулю);
- описание порядка применения и процедуры оценивания для каждого оценочного средства;
- критерии оценивания для каждого оценочного средства;
- содержание оценочных средств, включая требования, предъявляемые к действиям обучающихся, демонстрируемым результатам, задания различных типов.

Фонд оценочных средств по дисциплине находится в Приложении 1 к программе дисциплины (модулю).

## 7. Перечень литературы, необходимой для освоения дисциплины (модуля)

Освоение дисциплины (модуля) предполагает изучение основной и дополнительной учебной литературы. Литература может быть доступна обучающимся в одном из двух вариантов (либо в обоих из них):

- в электронном виде - через электронные библиотечные системы на основании заключенных КФУ договоров с правообладателями;
- в печатном виде - в Научной библиотеке им. Н.И. Лобачевского. Обучающиеся получают учебную литературу на абонементе по читательским билетам в соответствии с правилами пользования Научной библиотекой.

Электронные издания доступны дистанционно из любой точки при введении обучающимся своего логина и пароля от личного кабинета в системе "Электронный университет". При использовании печатных изданий библиотечный фонд должен быть укомплектован ими из расчета не менее 0,5 экземпляра (для обучающихся по ФГОС 3++ - не менее 0,25 экземпляра) каждого из изданий основной литературы и не менее 0,25 экземпляра дополнительной литературы на каждого обучающегося из числа лиц, одновременно осваивающих данную дисциплину.

Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины (модуля), находится в Приложении 2 к рабочей программе дисциплины. Он подлежит обновлению при изменении условий договоров КФУ с правообладателями электронных изданий и при изменении комплектования фондов Научной библиотеки КФУ.

## 8. Перечень ресурсов информационно-телекоммуникационной сети "Интернет", необходимых для освоения дисциплины (модуля)

Научная библиотека им. Н.И. Лобачевского КФУ - <https://library.kfu.ru/rus/el-resources/>

Портал "КФУ: Электронное образование" - <https://edu.kfu.ru/>

Электронная библиотечная система "Лань" - <https://e.lanbook.com/>

## 9. Методические указания для обучающихся по освоению дисциплины (модуля)

Вид работ	Методические рекомендации
практические занятия	Практические занятия - основная форма контактной работы. На них вы получаете практические навыки разработки компонентов распределенных систем. Перед занятием: изучите теоретический материал, проглядите примеры кода. На занятии: преподаватель продемонстрирует архитектуру и примеры, затем вы работаете самостоятельно над практическими заданиями. Разрабатывайте код, тестируйте его, используйте инструменты отладки. При возникновении проблем: проверьте логику, изучите логи, обратитесь к документации, обсудите с однокурсниками. Используйте Git для версионирования.
самостоятельная работа	Самостоятельная работа включает углублённое изучение теории, разработку приложений, анализ кейсов. Рекомендуемый алгоритм: (1) прочитайте конспект и материалы лекции/практики; (2) изучите соответствующие разделы в литературе; (3) посмотрите примеры реальных распределенных систем; (4) выполняйте практические упражнения; (5) разрабатывайте собственные приложения; (6) анализируйте production-системы (GitHub, открытые исходники). Используйте Moodle для доступа к материалам и консультаций.
экзамен	Экзамен оценивает освоение компетенции ПК-7. Подготовка: повторите все 9 тем, составьте конспекты, решите практические задачи. Формат: экзамен состоит из двух частей - теоретической (вопросы по архитектурам, протоколам, методам) и практической (разработка компонента распределенной системы). Демонстрируйте: понимание основных концепций, знание различных архитектур и их применения, умение выбрать подходящую архитектуру для задачи, практические навыки разработки. Ответ должен быть аргументированным, с примерами.

**10. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем (при необходимости)**

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), включая перечень программного обеспечения и информационных справочных систем, представлен в Приложении 3 к рабочей программе дисциплины (модуля).

**11. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю)**

Материально-техническое обеспечение образовательного процесса по дисциплине (модулю) включает в себя следующие компоненты:

Помещения для самостоятельной работы обучающихся, укомплектованные специализированной мебелью (столы и стулья) и оснащенные компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду КФУ.

Учебные аудитории для контактной работы с преподавателем, укомплектованные специализированной мебелью (столы и стулья).

Компьютер и принтер для распечатки раздаточных материалов.

**12. Средства адаптации преподавания дисциплины к потребностям обучающихся инвалидов и лиц с ограниченными возможностями здоровья**

При необходимости в образовательном процессе применяются следующие методы и технологии, облегчающие восприятие информации обучающимися инвалидами и лицами с ограниченными возможностями здоровья:

- создание текстовой версии любого нетекстового контента для его возможного преобразования в альтернативные формы, удобные для различных пользователей;
- создание контента, который можно представить в различных видах без потери данных или структуры, предусмотреть возможность масштабирования текста и изображений без потери качества, предусмотреть доступность управления контентом с клавиатуры;
- создание возможностей для обучающихся воспринимать одну и ту же информацию из разных источников - например, так, чтобы лица с нарушениями слуха получали информацию визуально, с нарушениями зрения - аудиально;
- применение программных средств, обеспечивающих возможность освоения навыков и умений, формируемых дисциплиной, за счёт альтернативных способов, в том числе виртуальных лабораторий и симуляционных технологий;
- применение дистанционных образовательных технологий для передачи информации, организации различных форм интерактивной контактной работы обучающегося с преподавателем, в том числе вебинаров, которые могут быть использованы для проведения виртуальных лекций с возможностью взаимодействия всех участников дистанционного обучения, проведения семинаров, выступления с докладами и защиты выполненных работ, проведения тренингов, организации коллективной работы;
- применение дистанционных образовательных технологий для организации форм текущего и промежуточного контроля;
- увеличение продолжительности сдачи обучающимся инвалидом или лицом с ограниченными возможностями здоровья форм промежуточной аттестации по отношению к установленной продолжительности их сдачи:
- продолжительности сдачи зачёта или экзамена, проводимого в письменной форме, - не более чем на 90 минут;
- продолжительности подготовки обучающегося к ответу на зачёте или экзамене, проводимом в устной форме, - не более чем на 20 минут;
- продолжительности выступления обучающегося при защите курсовой работы - не более чем на 15 минут.

Программа составлена в соответствии с требованиями ФГОС ВО и учебным планом по направлению 09.03.04 "Программная инженерия" и профилю подготовки "Современная разработка программного обеспечения".



*Приложение 2*  
*к рабочей программе дисциплины (модуля)*  
*Б1.В.ДВ.05.10 Распределенные вычисления и приложения*

**Перечень литературы, необходимой для освоения дисциплины (модуля)**

Направление подготовки: 09.03.04 - Программная инженерия

Профиль подготовки: Современная разработка программного обеспечения

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2025

**Основная литература:**

1. Филиппов, Ф. В. Многоядерные процессоры и параллельное программирование : учебное пособие / Ф. В. Филиппов, А. Н. Губин. - Санкт-Петербург : СПбГУТ им. М.А. Бонч-Бруевича, 2013. - 99 с. - Текст : электронный // Лань : электронно-библиотечная система. - URL: <https://e.lanbook.com/book/180060> (дата обращения: 10.12.2024). - Режим доступа: для авториз. пользователей.
2. Зорин, А. В. Методы Монте-Карло для параллельных вычислений : учебное пособие / Зорин А. В. , Федоткин М. А. - Москва : Издательство Московского государственного университета, 2013. - 192 с. (Серия 'Суперкомпьютерное образование') - ISBN 978-5-211-06530-7. - Текст : электронный // ЭБС 'Консультант студента' : [сайт]. - URL : <https://www.studentlibrary.ru/book/ISBN9785211065307.html> (дата обращения: 10.12.2024). - Режим доступа : по подписке.
3. Гримм, Р. Параллельное программирование на современном C++ / Р. Гримм; пер. с англ. В. Ю. Винника. - Москва : ДМК Пресс, 2022. - 616 с. - ISBN 978-5-97060-957-6. - Текст : электронный // ЭБС 'Консультант студента' : [сайт]. - URL : <https://www.studentlibrary.ru/book/ISBN9785970609576.html> (дата обращения: 10.12.2024). - Режим доступа : по подписке.

**Дополнительная литература:**

1. Максимов, Н. В. Архитектура ЭВМ и вычислительных систем : учебник / Н.В. Максимов, Т.Л. Партыка, И.И. Попов. - 5-е изд., перераб. и доп. - Москва : ФОРУМ : ИНФРА-М, 2017. - 511 с. - (Среднее профессиональное образование). - ISBN 978-5-16-104792-7. - Текст : электронный. - URL: <https://znanium.com/catalog/product/814513> (дата обращения: 10.12.2024). - Режим доступа : по подписке
2. Лупин, С. А. Технологии параллельного программирования : учебное пособие / С.А. Лупин, М.А. Посыпкин. - Москва : ИД 'ФОРУМ' : ИНФРА-М, 2021. - 206 с. - (Среднее профессиональное образование). - ISBN 978-5-8199-0853-2. - Текст : электронный. - URL: <https://znanium.ru/catalog/product/1189950> (дата обращения: 10.12.2024). - Режим доступа: по подписке.
3. Федотов, И. Е. Параллельное программирование. Модели и приемы : практическое пособие / И. Е. Федотов. - Москва : СОЛОН-Пресс, 2020. - 390 с. - (Серия 'Библиотека профессионала'). - ISBN 978-5-91359-222-4. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1858781> (дата обращения: 10.12.2024). - Режим доступа: по подписке.

*Приложение 3*  
*к рабочей программе дисциплины (модуля)*  
*Б1.В.ДВ.05.10 Распределенные вычисления и приложения*

**Перечень информационных технологий, используемых для освоения дисциплины (модуля), включая перечень программного обеспечения и информационных справочных систем**

Направление подготовки: 09.03.04 - Программная инженерия

Профиль подготовки: Современная разработка программного обеспечения

Квалификация выпускника: бакалавр

Форма обучения: очное

Язык обучения: русский

Год начала обучения по образовательной программе: 2025

Освоение дисциплины (модуля) предполагает использование следующего программного обеспечения и информационно-справочных систем:

Операционная система Microsoft Windows 7 Профессиональная или Windows XP (Volume License)

Пакет офисного программного обеспечения Microsoft Office 365 или Microsoft Office Professional plus 2010

Браузер Mozilla Firefox

Браузер Google Chrome

Adobe Reader XI или Adobe Acrobat Reader DC

Kaspersky Endpoint Security для Windows

Учебно-методическая литература для данной дисциплины имеется в наличии в электронно-библиотечной системе Издательства "Лань", доступ к которой предоставлен обучающимся. ЭБС Издательства "Лань" включает в себя электронные версии книг издательства "Лань" и других ведущих издательств учебной литературы, а также электронные версии периодических изданий по естественным, техническим и гуманитарным наукам. ЭБС Издательства "Лань" обеспечивает доступ к научной, учебной литературе и научным периодическим изданиям по максимальному количеству профильных направлений с соблюдением всех авторских и смежных прав.