Министерство науки и высшего образования Российской Федерации

Муромский институт (филиал)

федерального государственного бюджетного образовательного учреждения высшего образования

«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»

(МИ ВлГУ)

Кафедра *ИС*

«УТВЕРЖДАЮ»
Заместитель директора по УР
_____Д.Е. Андрианов
19.05.2026

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Основы облачных технологий

Направление подготовки *09.03.02 Информационные системы и технологии*

Профиль подготовки *Информационные системы и технологии*

Семестр	Трудоемкость, час./зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль, час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
6	**108 / 3**	**16**		**16**	**1,6**	**0,25**	**33,85**	**74,15**	**Зач.**
Итого	**108 / 3**	**16**		**16**	**1,6**	**0,25**	**33,85**	**74,15**	

Муром, 2026 г.

1. Цель освоения дисциплины

Цель:

Сформировать у студентов комплексное представление о современных облачных технологиях, включая принципы их работы, развертывания и управления, а также дать практические навыки работы с Docker-контейнерами в облачных средах для подготовки к профессиональной деятельности в области DevOps, cloud-разработки и системного администрирования.

Задачи:

1) Теоретические:

- Изучить базовые концепции облачных вычислений (IaaS, PaaS, SaaS) и модели развертывания (публичные, частные, гибридные облака).
- Рассмотреть принципы контейнеризации, архитектуру Docker и его роль в облачных технологиях.
- Изучить вопросы безопасности, масштабируемости и экономической эффективности облачных решений.

2) Практические:

- Освоить работу с ведущими облачными платформами (AWS, Azure, GCP) – развертывание виртуальных машин, настройка сетей, управление ресурсами.
- Научиться создавать, запускать и управлять Docker-контейнерами, работать с Docker-образами и оркестрацией (Docker Compose).
- Приобрести навыки интеграции Docker-контейнеров с облачными сервисами для развертывания отказоустойчивых приложений.

3) Прикладные:

- Развить умение проектировать и оптимизировать облачную инфраструктуру с использованием контейнеризации.
- Научиться применять инструменты мониторинга и управления облачными ресурсами.
- Сформировать навыки командной работы над проектами с применением облачных технологий и Docker.

2. Место дисциплины в структуре ОПОП ВО

Курс базируется на дисциплинах: Операционные системы, Инфокоммуникационные системы и сети. На данном курсе основаны: Моделирование информационных систем, Информационная безопасность, Тестирование информационных систем.

3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
ПК-5 Способность выполнять работы по созданию (модификации) и сопровождению информационных систем	ПК-5.1 Применяет современные средства создания, модификации и сопровождения информационных систем	- ключевые принципы работы облачных платформ и модели развертывания: публичные, частные, гибридные облака (ПК-5.1) - основы контейнеризации, включая архитектуру Docker, его преимущества и сферы применения в облачных средах (ПК-5.1) - развертывать и управлять облачной инфраструктурой с использованием	вопросы устного опроса, задания для лабораторных работ

		популярных платформ: AWS, Azure, GCP (ПК-5.1) - создавать, запускать и масштабировать Docker-контейнеры, а также работать с Docker-образами и Docker Compose (ПК-5.1)	
ПК-1 Способность проводить исследования на всех этапах жизненного цикла программных средств	ПК-1.1 Применяет основные подходы исследования на всех этапах жизненного цикла программых средств	- настройкой виртуальных машин, контейнеров и бессерверных сервисов в облаке (ПК-1.1) - проектирования отказоустойчивых и безопасных облачных решений с применением Docker (ПК-1.1) - мониторинга и оптимизации облачных ресурсов для эффективного использования (ПК-1.1)	вопросы устного опроса, задания для лабораторных работ

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 3 зачетные единицы, 108 часов.

4.1. Форма обучения: очная

Уровень базового образования: среднее общее.
Срок обучения 4г.

4.1.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Введение в основы облачных технологий	6	6		4					46	устный опрос, отчеты по лабораторным работам
2	Использование технологии Docker	6	10		12					28,15	устный опрос, отчеты по лабораторным работам
Всего за семестр		108	16		16			1,6	0,25	74,15	Зач.
Итого		108	16		16			1,6	0,25	74,15	

4.1.2. Содержание дисциплины

4.1.2.1. Перечень лекций

Семестр 6

Раздел 1. Введение в основы облачных технологий

Лекция 1.

Введение в дисциплину. Предпосылки. Основные понятия (2 часа).

Лекция 2.

Виртуализация (2 часа).

Лекция 3.

Контейнеризация (2 часа).

Раздел 2. Использование технологии Docker

Лекция 4.

Docker-образы (2 часа).

Лекция 5.

Масштабирование и безопасность (2 часа).

Лекция 6.

Оркестрация (2 часа).

Лекция 7.

Архитектура облачных приложений (2 часа).

Лекция 8.

Введение в CI/CD (2 часа).

4.1.2.2. Перечень практических занятий

Не планируется.

4.1.2.3. Перечень лабораторных работ

Семестр 6

Раздел 1. Введение в основы облачных технологий

Лабораторная 1.

Использование chroot и LXC (4 часа).

Раздел 2. Использование технологии Docker

Лабораторная 2.

Основы использования docker и podman (4 часа).

Лабораторная 3.

Многоэтапная сборка docker-образов (4 часа).

Лабораторная 4.

Использование docker-compose (4 часа).

4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Сравнительный анализ IaaS, PaaS, SaaS: преимущества и ограничения.
2. Принципы работы гипервизоров (Type 1 и Type 2): KVM, Hyper-V, VMware.
3. Альтернативные технологии контейнеризации: LXC, rkt, containerd.
4. Основы Kubernetes: Pods, Deployments, Services.
5. Автомасштабирование в облаке: Horizontal Pod Autoscaler (HPA) и Cloud Auto Scaling.
6. Облачная безопасность: IAM, шифрование данных, защита от DDoS.
7. Serverless-архитектура: AWS Lambda, Azure Functions, Google Cloud Functions.
8. Микросервисы vs. монолит: плюсы и минусы архитектур.
9. Основы CI/CD: Jenkins, GitLab CI, GitHub Actions.
10. Инфраструктура как код (IaC): Terraform, Ansible, Pulumi.
11. Мониторинг облачных приложений: Prometheus, Grafana, CloudWatch.
12. Docker vs. Podman: сходства, различия и сценарии использования.
13. Оптимизация Docker-образов: уменьшение размера и ускорение сборки.
14. Работа с Docker Hub и альтернативными реестрами (GitHub Container Registry, AWS ECR).
15. Docker Volumes и Bind Mounts: управление данными в контейнерах.
16. Docker Networking: мосты, overlay-сети и настройка взаимодействия контейнеров.
17. Безопасность Docker: best practices, сканирование образов (Trivy, Clair).
18. Kubernetes vs. Docker Swarm: сравнение оркестраторов.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

Не планируется.

4.1.2.6. Примерный перечень тем курсовых работ (проектов)

Не планируется.

4.2 Форма обучения: заочная

Уровень базового образования: среднее профессиональное.
Срок обучения 3г 6м.

Семестр	Трудоемкость, час./ зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль,час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
6	**108 / 3**	**4**		**4**	**2**	**0,5**	**10,5**	**93,75**	**Зач.(3,75)**
Итого	**108 / 3**	**4**		**4**	**2**	**0,5**	**10,5**	**93,75**	**3,75**

4.2.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Введение в основы облачных технологий	6	4							64	Устный опрос
2	Использование технологии Docker	6			4					29,75	Устный опрос, лабораторная работа
Всего за семестр		108	4		4	+		2	0,5	93,75	Зач.(3,75)
Итого		108	4		4			2	0,5	93,75	3,75

4.2.2. Содержание дисциплины

4.2.2.1. Перечень лекций

Семестр 6

Раздел 1. Введение в основы облачных технологий

Лекция 1.

Введение в дисциплину. Предпосылки. Основные понятия (2 часа).

Лекция 2.

Контейнеризация (2 часа).

4.2.2.2. Перечень практических занятий

Не планируется.

4.2.2.3. Перечень лабораторных работ

Семестр 6

Раздел 1. Использование технологии Docker

Лабораторная 1.

Основы использования docker и podman (4 часа).

4.2.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Сравнительный анализ IaaS, PaaS, SaaS: преимущества и ограничения.
2. Принципы работы гипервизоров (Type 1 и Type 2): KVM, Hyper-V, VMware.
3. Альтернативные технологии контейнеризации: LXC, rkt, containerd.
4. Основы Kubernetes: Pods, Deployments, Services.
5. Автомасштабирование в облаке: Horizontal Pod Autoscaler (HPA) и Cloud Auto Scaling.
6. Облачная безопасность: IAM, шифрование данных, защита от DDoS.
7. Serverless-архитектура: AWS Lambda, Azure Functions, Google Cloud Functions.
8. Микросервисы vs. монолит: плюсы и минусы архитектур.
9. Основы CI/CD: Jenkins, GitLab CI, GitHub Actions.
10. Инфраструктура как код (IaC): Terraform, Ansible, Pulumi.
11. Мониторинг облачных приложений: Prometheus, Grafana, CloudWatch.
12. Docker vs. Podman: сходства, различия и сценарии использования.
13. Оптимизация Docker-образов: уменьшение размера и ускорение сборки.
14. Работа с Docker Hub и альтернативными реестрами (GitHub Container Registry, AWS ECR).
15. Docker Volumes и Bind Mounts: управление данными в контейнерах.
16. Docker Networking: мосты, overlay-сети и настройка взаимодействия контейнеров.
17. Безопасность Docker: best practices, сканирование образов (Trivy, Clair).
18. Kubernetes vs. Docker Swarm: сравнение оркестраторов.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.2.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

1. Эволюция облачных технологий: от мейнфреймов до современных облачных платформ.
2. Модели обслуживания в облаке (IaaS, PaaS, SaaS): сравнительный анализ.
3. Публичные, частные и гибридные облака: преимущества и недостатки.
4. Мультиоблачные стратегии: зачем компании используют несколько провайдеров?.
5. Edge computing и его интеграция с облачными технологиями.
6. Гипервизоры Type 1 и Type 2: KVM, Hyper-V, VMware ESXi.
7. Контейнеры vs. виртуальные машины: производительность и сценарии использования.
8. История развития контейнеризации: от chroot и LXC до Docker.
9. Docker и Podman: сравнение архитектуры и функциональности.
10. Альтернативные системы контейнеризации: rkt, containerd, CRI-O.
11. Docker-образы: структура, слои, оптимизация размера.
12. Многоэтапная сборка Docker-образов: зачем она нужна и как ее использовать?.
13. Docker Hub и альтернативные реестры контейнеров (GitHub Container Registry, AWS ECR).
14. Управление данными в Docker: Volumes, Bind Mounts и tmpfs.
15. Docker Networking: bridge, host, overlay и macvlan-драйверы.
16. Docker Swarm: архитектура и базовые принципы работы.
17. Kubernetes: основные концепции (Pods, Deployments, Services).

18. Сравнение Kubernetes и Docker Swarm: что выбрать?.
19. Helm: система управления пакетами в Kubernetes.
20. Сервис-меши (Service Mesh): Istio, Linkerd, Consul.
21. Amazon Web Services (AWS): основные сервисы и архитектура.
22. Microsoft Azure: особенности и сравнение с AWS.
23. Google Cloud Platform (GCP): ключевые преимущества.
24. Бессерверные вычисления: AWS Lambda vs. Azure Functions vs. Google Cloud Functions.
25. Облачные базы данных: Amazon RDS, Azure SQL, Google Cloud Spanner.
26. Модель общей ответственности в облачных сервисах.
27. Identity and Access Management (IAM) в AWS, Azure, GCP.
28. Шифрование данных в облаке: at-rest и in-transit.
29. Защита от DDoS-атак в облачных средах.
30. Аудит безопасности: AWS CloudTrail, Azure Monitor, Google Cloud Audit Logs.
31. Auto Scaling в облаке: горизонтальное и вертикальное масштабирование.
32. Балансировка нагрузки: AWS ALB, Azure Load Balancer, GCP Cloud Load Balancing.
33. Геораспределенные приложения: как обеспечить низкие задержки?.
34. Disaster Recovery в облаке: стратегии и лучшие практики.
35. High Availability (HA) архитектуры в Kubernetes.
36. Непрерывная интеграция и доставка (CI/CD): принципы и инструменты.
37. GitLab CI/CD vs. GitHub Actions vs. Jenkins: что выбрать?.
38. Инфраструктура как код (IaC): Terraform, Pulumi, AWS CDK.
39. Управление конфигурациями: Ansible, Chef, Puppet.
40. Мониторинг и логирование: Prometheus, Grafana, ELK Stack.
41. Монолит vs. микросервисы: когда что использовать?.
42. Serverless-архитектура: плюсы и минусы.
43. Event-Driven Architecture в облаке (Kafka, AWS SQS, Azure Event Grid).
44. Кэширование в облачных приложениях: Redis, Memcached, CDN.
45. Оптимизация затрат в облаке: как избежать лишних расходов?.
46. Как Netflix использует облачные технологии?.
47. Облачные решения для Big Data: AWS EMR, Azure HDInsight, Google Dataproc.
48. Искусственный интеллект и машинное обучение в облаке (AWS SageMaker, Azure ML).
49. Квантовые вычисления и облака: перспективы развития.
50. Green Cloud Computing: как снизить энергопотребление дата-центров?.

4.2.2.6. Примерный перечень тем курсовых работ (проектов)

Не планируется.

5. Образовательные технологии

В процессе изучения дисциплины применяется контактная технология преподавания (за исключением самостоятельно изучаемых студентами вопросов). При проведении лабораторных работ применяется имитационный или симуляционный подход. Шаги решения задач студентам демонстрируются при помощи мультимедийной техники. В дальнейшем студенты самостоятельно решают аналогичные задания.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Фонды оценочных материалов (средств) приведены в приложении.

7. Учебно-методическое и информационное обеспечение дисциплины.

7.1. Основная учебно-методическая литература по дисциплине

1. Ларина, Т. Б. Виртуализация операционных систем : учебное пособие / Т. Б. Ларина. — Москва : Российский университет транспорта (МИИТ), 2020. — 65 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/115824.html. — Режим доступа: для авторизир. пользователей - https://www.iprbookshop.ru/115824.html

2. Инфраструктура и архитектура виртуализации : учебное пособие / составители И. А. Ботыгин, В. С. Шерстнев, А. И. Шерстнева. — Томск : Томский политехнический университет, 2021. — 224 с. — ISBN 978-5-4387-1045-5. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/134276.html. — Режим доступа: для авторизир. пользователей - https://www.iprbookshop.ru/134276.html

7.2. Дополнительная учебно-методическая литература по дисциплине

1. Клашанов, Ф. К. Вычислительные системы и сети, облачные технологии : учебно-методическое пособие / Ф. К. Клашанов. — Москва : МИСИ-МГСУ, ЭБС АСВ, 2020. — 40 с. — ISBN 978-5-7264-2187-2. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/101788.html (дата обращения: 26.03.2025). — Режим доступа: для авторизир. пользователей - https://www.iprbookshop.ru/101788.html

2. Зиангирова, Л. Ф. Облачные вычисления : учебное пособие / Л. Ф. Зиангирова. — 2-е изд. — Москва : Ай Пи Ар Медиа, 2024. — 168 с. — ISBN 978-5-4497-3428-0. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/142083.html (дата обращения: 26.03.2025). — Режим доступа: для авторизир. пользователей - https://www.iprbookshop.ru/142083.html

3. Савельев, А. О. Введение в облачные решения Microsoft : учебное пособие / А. О. Савельев. — 4-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 229 с. — ISBN 978-5-4497-0877-9. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/146331.html (дата обращения: 26.03.2025). — Режим доступа: для авторизир. пользователей - https://www.iprbookshop.ru/146331.html

4. Губарев, В. В. Введение в облачные вычисления и технологии : учебное пособие / В. В. Губарев, С. А. Савульчик, Н. А. Чистяков. — Новосибирск : Новосибирский государственный технический университет, 2013. — 48 с. — ISBN 978-5-7782-2252-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/44905.html (дата обращения: 26.03.2025). — Режим доступа: для авторизир. пользователей - https://www.iprbookshop.ru/44905.html

5. Зиангирова, Л. Ф. Технологии облачных вычислений : учебное пособие для СПО / Л. Ф. Зиангирова. — 2-е изд. — Саратов, Москва : Профобразование, Ай Пи Ар Медиа, 2024. — 252 с. — ISBN 978-5-4488-2175-2, 978-5-4497-3426-6. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/142233.html (дата обращения: 26.03.2025). — Режим доступа: для авторизир. пользователей - https://www.iprbookshop.ru/142233.html

7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института (www.mivlgu.ru/iop), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;

- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

База знаний по продуктам компании Ред Софт (https://redos.red-soft.ru/base/)

Портал документации по продуктам компании Ред Софт (https://redos.red-soft.ru/product/docs/)

Портал технической информации Habr (https://habr.com/ru/)

Программное обеспечение:

LibreOffice (Mozilla Public License v2.0)

7-Zip (GNU LGPL)

Google Chrome (Лицензионное соглашение Google)

Mozilla Firefox (MPL)

РЕД ОС (Соглашение №140/05-21У от 18.05.2021 года о сотрудничестве в области науки, развития инновационной деятельности)

Foxit Reader (Foxit EULA)

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

iprbookshop.ru

mivlgu.ru/iop

8. Материально-техническое обеспечение дисциплины

Компьютерный класс

Проектор ViewSonic PG603X DLP Экран Lumien Персональный компьютер RUSCO – 19 шт. Коммутатор D-Link Маршрутизатор беспроводной N ASUS RT-AC66U

9. Методические указания по освоению дисциплины

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями.

До выполнения лабораторных работ обучающийся изучает соответствующий раздел теории. Перед занятием студент знакомится с описанием заданий для выполнения работы, внимательно изучает содержание и порядок проведения лабораторной работы. Лабораторная работа проводятся в компьютерном классе. Обучающиеся выполняют индивидуальную задачу компьютерного моделирования в соответствии с заданием на лабораторную работу. Полученные результаты исследований сводятся в отчет и защищаются по традиционной методике в классе на следующем лабораторном занятии. Необходимый теоретический материал, индивидуальное задание, шаги выполнения лабораторной работы и требование к отчету приведены в методических указаниях, размещенных на информационно-образовательном портале института.

Самостоятельная работа оказывает важное влияние на формирование личности будущего специалиста, она планируется обучающимся самостоятельно. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Форма заключительного контроля при промежуточной аттестации – зачет. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению *09.03.02 Информационные системы и технологии* и профилю подготовки *Информационные системы и технологии*
Рабочую программу составил *ст. преподаватель Булаев А.В.*_____

Программа рассмотрена и одобрена на заседании кафедры *ИС*

протокол № 21 от 05.05.2026 года.
Заведующий кафедрой *ИС* _____*Андрианов Д.Е.*
(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии факультета

протокол № 9 от 12.05.2026 года.
Председатель комиссии ФИТР _____ *Кутарова Е.И.*
(Подпись) (Ф.И.О.)

Приложение

Фонд оценочных материалов (средств) по дисциплине

Основы облачных технологий

1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине

Вопросы для текущего контроля знаний:

1. Какое определение облачных технологий дается в лекции?
2. Какие три вида облаков выделяются по модели развертывания?
3. Какие преимущества использования облачных технологий отмечаются в исследованиях Accenture?
4. Что такое IaaS, и кому она может быть полезна?
5. Какие основные модели облачных технологий существуют?
6. Какие плюсы и минусы облачных технологий перечислены в лекции?
7. Как развивались облачные платформы с 1960-х годов до 2010-х?
8. Какие сферы применения облачных платформ упоминаются в лекции?
9. Как облачные технологии используются в образовании?
10. Какие перспективы развития облачных технологий отмечаются в лекции?
11. Какая проблема привела к появлению концепции виртуализации?
12. Какие компоненты входили в систему CP-67/CMS?
13. Какие преимущества виртуализации над концепцией разделения времени?
14. Какие продукты для виртуализации появились в 1990-х и 2000-х годах?
15. Какие три основных вида виртуализации выделяются?
16. В чем разница между аппаратной виртуализацией и виртуализацией рабочих столов?
17. Что такое гипервизор, и какие типы гипервизоров существуют?
18. Каковы особенности гипервизоров первого типа?
19. Чем отличаются гипервизоры второго типа от гибридных?
20. Какие задачи решает виртуализация в современной IT-инфраструктуре?
21. В чем основное отличие контейнеризации от виртуализации?
22. Какие преимущества контейнеров перед виртуальными машинами?
23. Что такое образ контейнера, и как он создается?
24. Какие механизмы ядра Linux используются для изоляции контейнеров?
25. Что такое многоконтейнерные приложения, и какие преимущества они предоставляют?
26. Какие задачи решают оркестраторы контейнеров?
27. Какие оркестраторы контейнеров упоминаются в лекции?
28. Что такое микросервисная архитектура, и как она связана с контейнеризацией?
29. Какие преимущества микросервисной архитектуры?
30. Как контейнеризация упрощает процессы CI/CD?
31. Что такое Docker, и какова его архитектура?
32. Какие основные компоненты контейнеризации упоминаются в лекции?
33. Чем отличаются Docker, Podman и containerd?
34. Что такое OCI, и какие стандарты он включает?
35. Как создается Docker-образ, и какую роль играет Dockerfile?
36. Какие основные команды используются в Dockerfile?
37. В чем разница между Docker-образом и Docker-контейнером?
38. Какие механизмы хранения данных в Docker упоминаются?
39. Что такое Docker-реестр, и какие виды реестров существуют?
40. Как обеспечивается версионность Docker-образов?

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	устный опрос, отчеты по лабораторным работам	20
Рейтинг-контроль 2	устный опрос, отчеты по лабораторным работам	20
Рейтинг-контроль 3	устный опрос, отчеты по лабораторным работам	20
Посещение занятий студентом	конспект лекций	10
Дополнительные баллы (бонусы)		10
Выполнение семестрового плана самостоятельной работы	отчет по самостоятельной работе	20

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.
Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

Перечень вопросов, выносимых на итоговую аттестацию.

1. Что такое облачные технологии и каковы их основные преимущества?
2. Назовите три основные модели развертывания облаков и дайте их краткую характеристику.
3. В чем разница между публичным, частным и гибридным облаком?
4. Какие основные сервисные модели облачных технологий вы знаете? Опишите каждую.
5. Приведите примеры использования облачных технологий в бизнесе.
6. Что такое виртуализация и какие основные виды виртуализации существуют?
7. Какие преимущества дает виртуализация серверов?
8. Опишите различия между гипервизорами первого и второго типа.
9. Какие задачи решает виртуализация рабочих столов?
10. Назовите популярные решения для виртуализации и их особенности.
11. Что такое контейнеризация и чем она отличается от виртуализации?
12. Какие преимущества предоставляет использование контейнеров?
13. Опишите основные компоненты Docker.
14. Что такое Docker-образ и как он создается?
15. Для чего используется Docker Compose?
16. Какие основные угрозы безопасности существуют для контейнерных технологий?
17. Как можно минимизировать привилегии контейнеров для повышения безопасности?
18. Что такое Docker Secrets и для чего они используются?
19. Какие инструменты используются для сканирования уязвимостей в контейнерах?
20. Как обеспечить безопасность данных в облачных хранилищах?
21. Что такое оркестрация контейнеров и зачем она нужна?
22. Назовите основные задачи, которые решают оркестраторы.
23. В чем разница между Kubernetes и Docker Swarm?
24. Какие компоненты входят в архитектуру Kubernetes?
25. Как оркестраторы помогают в масштабировании приложений?
26. Что такое микросервисная архитектура и каковы ее преимущества?

27. В чем разница между Serverless и Cloud Native подходами?
28. Какие компоненты входят в Serverless-архитектуру на примере AWS?
29. Как Cloud Native подход влияет на разработку приложений?
30. Какие проблемы могут возникнуть при переходе на Cloud Native архитектуру?
31. Что такое CI/CD и какие этапы включает этот процесс?
32. Какие преимущества дает внедрение CI/CD в разработку?
33. Опишите основные инструменты для реализации CI/CD.
34. Как CI/CD интегрируется с Kubernetes?
35. Что такое пайплайн в GitLab CI/CD и как он настраивается?
36. История и развитие облачных технологий
37. Какие этапы развития прошли облачные технологии?
38. Как изменились облачные технологии за последние 10 лет?
39. Какие компании являются лидерами в предоставлении облачных услуг?
40. Как пандемия повлияла на развитие облачных технологий?
41. Какие перспективы развития облачных технологий вы видите?
42. Применение облачных технологий в различных сферах
43. Как облачные технологии используются в образовании?
44. Приведите примеры использования облаков в медицине.
45. Как банковский сектор использует облачные технологии?
46. Какие облачные решения популярны в логистике?
47. Как облачные технологии помогают в управлении бизнес-процессами?
48. Как создать и запустить контейнер с помощью Docker?
49. Опишите процесс настройки простого CI/CD-пайплайна в GitLab.
50. Как развернуть многоконтейнерное приложение с помощью Docker Compose?
51. Какие команды используются для управления контейнерами в Docker?
52. Как обеспечить отказоустойчивость приложения с помощью Kubernetes?

Методические материалы, характеризующие процедуры оценивания

Промежуточная аттестация проходит в форме устного опроса, в ходе которого студент дает ответ на вопросы билета. Итоговоая оценка формируется на основе ответа студента на вопросы билета, дополнительные и уточняющие вопросы преподаватели и с учетом выполнения студентом лабораторных работ в течении учебного семестра.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	***Уровень сформированности компетенций***
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	***Высокий уровень***
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все	***Продвинутый уровень***

		предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	
50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	***Пороговый уровень***
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	***Компетенции не сформированы***

3. Задания в тестовой форме по дисциплине

Примеры заданий:

1. Какой тип гипервизора работает непосредственно на аппаратном обеспечении без хостовой ОС?

A) Гипервизор второго типа
B) Гибридный гипервизор
C) Гипервизор первого типа
D) Контейнерный гипервизор

2. Какой проект IBM стал основой для современных технологий виртуализации?

A) VMware ESXi
B) CP-40/CMS
C) Xen
D) VirtualBox

3. Какой механизм виртуализации позволяет запускать несколько ОС на одном физическом сервере с изоляцией процессов?

A) Контейнеризация
B) Аппаратная виртуализация
C) Эмуляция
D) Паравиртуализация

4. Какой из перечисленных гипервизоров относится ко второму типу?

A) VMware ESXi
B) KVM
C) Oracle VM VirtualBox
D) Microsoft Hyper-V

5. Какой компонент в архитектуре виртуализации отвечает за управление виртуальными машинами?

A) Гостевая ОС
B) Гипервизор
C) Хостовая ОС
D) Драйвер устройств

6. Чем контейнеры отличаются от виртуальных машин?

A) Контейнеры используют отдельное ядро ОС

B) Контейнеры работают только в Windows

C) Контейнеры используют общее ядро хостовой ОС

D) Контейнеры требуют гипервизор

7. Какой компонент Docker используется для хранения изменений в контейнере?

A) Dockerfile

B) Docker Volume

C) Docker Hub

D) Docker Compose

8. Какой командой можно собрать Docker-образ из Dockerfile?

A) docker run

B) docker pull

C) docker build

D) docker create

9. Какой инструмент используется для управления многоконтейнерными приложениями в Docker?

A) Kubernetes

B) Docker Swarm

C) Docker Compose

D) Nomad

10. Какой механизм в Docker позволяет изолировать процессы внутри контейнера?

A) Виртуализация CPU

B) Namespaces и Cgroups

C) Гипервизор

D) Эмуляция памяти

11. Какой оркестратор использует концепцию "подов" (pods)?

A) Docker Swarm

B) Kubernetes

C) Nomad

D) Apache Mesos

12. Какой компонент Kubernetes отвечает за выбор узла для запуска пода?

A) kubelet

B) kube-proxy

C) kube-scheduler

D) etcd

13. Какой стратегии обновления нет в Kubernetes?

A) Rolling Update

B) Blue/Green

C) Recreate

D) Cold Start

14. Какой инструмент позволяет управлять секретами в Kubernetes?

A) ConfigMap

B) Secrets

C) Vault

D) Ansible

15. Какой оркестратор разработан HashiCorp и поддерживает не только контейнеры?

A) Kubernetes

B) Docker Swarm

C) Nomad

D) OpenShift

16. Какой инструмент используется для сканирования уязвимостей в Docker-образах?

A) Docker Scout

B) Trivy
C) Nessus
D) Wireshark
17. Какой механизм в Docker позволяет ограничить системные вызовы контейнера?
A) SELinux
B) AppArmor
C) Seccomp
D) Firewall
18. Какой параметр в Docker ограничивает привилегии контейнера?
A) --privileged
B) --cap-drop
C) --net=host
D) --volume
19. Какой сервис AWS позволяет управлять секретами в облаке?
A) AWS Secrets Manager
B) AWS KMS
C) AWS IAM
D) AWS S3
20. Какой атаке подвержены контейнеры, если они запущены с правами root?
A) DDoS
B) Escape-атака
C) MITM
D) SQL-инъекция
21. Какой сервис AWS является бессерверной вычислительной платформой?
A) EC2
B) Lambda
C) ECS
D) S3
22. Какой подход предполагает автоматическое масштабирование ресурсов под нагрузку?
A) IaaS
B) PaaS
C) Serverless
D) SaaS
23. Какой компонент AWS используется для маршрутизации событий между сервисами?
A) SQS
B) SNS
C) EventBridge
D) API Gateway
24. Какой сервис в AWS предоставляет бессерверную базу данных?
A) RDS
B) DynamoDB
C) Redshift
D) Aurora
25. Какой принцип Cloud Native предполагает использование микросервисов?
A) Монолитная архитектура
B) Гибридная архитектура
C) Распределенная архитектура
D) Серверная архитектура
26. Какой этап CI/CD отвечает за автоматическое развертывание кода?
A) Continuous Integration
B) Continuous Deployment
C) Continuous Testing

D) Continuous Monitoring

27. Какой инструмент НЕ является CI/CD-решением?

A) Jenkins

B) GitLab CI

C) Ansible

D) CircleCI

28. Какой файл описывает пайплайн в GitLab CI?

A) Dockerfile

B) .gitlab-ci.yml

C) Jenkinsfile

D) pipeline.yaml

29. Какой подход позволяет временно отключать новый функционал в продакшене?

A) Blue/Green

B) Feature Flags

C) Canary Release

D) Rolling Update

30. Какой сервис AWS используется для автоматизации развертывания инфраструктуры?

A) CloudFormation

B) CodeDeploy

C) CodePipeline

D) Elastic Beanstalk

31. Какой объект Kubernetes используется для масштабирования приложений на основе метрик CPU?

A) Deployment

B) HorizontalPodAutoscaler (HPA)

C) StatefulSet

D) DaemonSet

32. Какой компонент Kubernetes хранит состояние кластера?

A) kube-apiserver

B) etcd

C) kube-controller-manager

D) kube-scheduler

33. Какой ресурс Kubernetes ограничивает использование CPU для пода?

A) requests

B) limits

C) quotas

D) taints

34. Какой механизм Kubernetes позволяет обновлять приложения без downtime?

A) RollingUpdate

B) Recreate

C) Blue/Green Deployment

D) Canary Release

35. Какой объект Kubernetes используется для управления сетевым трафиком между подами?

A) Service

B) Ingress

C) NetworkPolicy

D) Pod

36. Какой протокол НЕ используется для VPN-подключения к облачным провайдерам?

A) IPsec

B) OpenVPN

C) HTTP

D) WireGuard

37. Какой сервис AWS обеспечивает изоляцию сетевых ресурсов?

A) VPC

B) S3

C) Lambda

D) CloudFront

38. Какой компонент Kubernetes отвечает за балансировку нагрузки?

A) kube-proxy

B) CoreDNS

C) Ingress Controller

D) etcd

39. Какой инструмент используется для управления сетевыми политиками в Kubernetes?

A) Calico

B) Flannel

C) Istio

D) Envoy

40. Какой тип балансировки нагрузки работает на 4-м уровне модели OSI?

A) L7 Load Balancer

B) L4 Load Balancer

C) DNS Load Balancer

D) CDN

41. Какой инструмент НЕ является системой мониторинга?

A) Prometheus

B) Grafana

C) ELK Stack

D) Terraform

42. Какой формат метрик используется в Prometheus?

A) JSON

B) XML

C) Key-Value

D) Text-based (PromQL)

43. Какой компонент Kubernetes собирает логи контейнеров?

A) kubelet

B) Fluentd

C) kube-proxy

D) CoreDNS

44. Какой сервис AWS используется для централизованного сбора логов?

A) CloudTrail

B) CloudWatch Logs

C) S3

D) Lambda

45. Какой инструмент позволяет визуализировать метрики из Prometheus?

A) Kibana

B) Grafana

C) Splunk

D) Datadog

46. Какой инструмент НЕ относится к IaC?

A) Terraform

B) Ansible

C) Docker

D) Puppet

47. Какой язык используется для написания конфигураций в Terraform?

A) YAML

B) JSON
C) HCL
D) Python

48. Какой команды НЕТ в Terraform?
A) terraform apply
B) terraform destroy
C) terraform build
D) terraform plan

49. Какой инструмент AWS позволяет развертывать инфраструктуру через JSON/YAML-шаблоны?
A) CloudFormation
B) CodeDeploy
C) Elastic Beanstalk
D) OpsWorks

50. Какой принцип IaC предполагает неизменяемость инфраструктуры?
A) Идемпотентность
B) Immutable Infrastructure
C) Declarative Configuration
D) Infrastructure Drift

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке https://www.mivlgu.ru/iop/question/edit.php?courseid=4260&cat=22635%2C238&qpage=0&deleteall=1&category=70805%2C228791&qbshowtext=0&recurse=0&recurse=1&showhidden=0&showhidden=1

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.