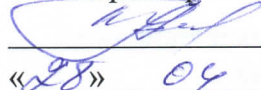


МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ
И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ
Северо-Кавказский филиал
ордена Трудового Красного Знамени федерального государственного
бюджетного образовательного учреждения высшего образования
«Московский технический университет связи и информатики»

Утверждаю

Зам. директора по УР



Н.А. Андреева

«28» 04 2026 г.

ОСНОВЫ DEVOPS Б1.В.05
рабочая программа дисциплины

Кафедра **Информатики и вычислительной техники**

Направление подготовки **09.03.01. Информатика и вычислительная техника**

Профиль **Искусственный интеллект и машинное обучение**

Формы обучения **очная, заочная**

**Распределение часов дисциплины по семестрам (для очной формы обучения (ОФО)),
курсам (для заочной формы обучения (ЗФО))**

Вид учебной работы	ОФО		ЗФО	
	ЗЕ	часов/сем.	ЗЕ	часов/курс
Общая трудоемкость дисциплины, в том числе (по семестрам, курсам):	4	144/5	4	144/3
Контактная работа, в том числе (по семестрам, курсам):		64/5		22/3
Лекции		16/5		8/3
Лабораторных работ		32/5		8/3
Практических занятий		16/5		6/3
Семинаров				
Самостоятельная работа		44/5		113/3
Контроль		36/5		9/3
Число контрольных работ (по курсам)				
Число КР (по семестрам, курсам)				
Число КП (по семестрам, курсам)				
Число зачетов с разбивкой по семестрам				
Число экзаменов с разбивкой по семестрам (курсам)		1/5		1/3

Программу составил:

ст. преподаватель кафедры ИВТ Коноваленко А.В.

Рабочая программа дисциплины
«Основы DevOps»

Разработана в соответствии с ФГОС ВО:

направления подготовки 09.03.01 Информатика и вычислительная техника, утвержден-
ным приказом Министерства науки и высшего образования Российской Федерации от 19
сентября 2017 г. N 929

Составлена на основании учебных планов

направления 09.03.01 Информатика и вычислительная техника

профиль "Искусственный интеллект и машинное обучение", одобренных Учёным советом
СКФ МТУСИ, протокол № 8 от 31.03. 2026г, и утвержденного директором
СКФ МТУСИ 31.03. 2026 г.

Рассмотрена и одобрена на заседании кафедры

"Информатика и вычислительная техника"

Протокол от «28» 04 2026 г. № 9

Зав. кафедрой  Соколов С.В.

1. Цели изучения дисциплины

Целью дисциплины является профессиональная подготовка студентов, необходимая для освоения методов и технологий разработки современных баз данных, являющихся основой информационных систем в различных сферах профессиональной деятельности.

2. Планируемые результаты обучения

Изучение данной учебной дисциплины направлено на формирование у обучающихся компетенций, представленных в следующей таблице:

Компетенции обучающегося, формируемые в результате освоения дисциплины (в части, обеспечиваемой дисциплиной)	
ПК-3: Способен разрабатывать и внедрять программное обеспечение, осуществлять его поддержку	
Знать:	
– методы разработки, внедрения и сопровождения программного обеспечения	
Уметь:	
– разрабатывать программное обеспечение с применением современных средств и технологий в соответствии с требованиями; – выполнять внедрение и сопровождение программного обеспечения; – выбирать эффективные алгоритмы решения задач профессиональной деятельности.	
Владеть:	
– приемами эффективного решения задач в рамках профессиональной деятельности	

3. Место дисциплины в структуре образовательной программы

Требования к предварительной подготовке обучающегося (предшествующие дисциплины, модули, темы):	
1	Б1.О.10 «Математические основы баз данных»
2	Б1.О.30 «Структуры и алгоритмы обработки данных»
3	Б1.О.13 «Операционные системы»
Последующие дисциплины и практики, для которых освоение данной дисциплины необходимо:	
1	Б1.В.ДВ.01.01 «Методы отладки и тестирования программного обеспечения»
2	Б1.В.ДВ.01.02 «Распределенные вычисления»
3	Б1.В.ДВ.05.01 «Разработка кроссплатформенных приложений»
4	Б3.01 Государственная итоговая аттестация

4. Структура и содержание дисциплины

4.1. Очная форма обучения, 4 года (всего 144 часов, из них 64 часов аудиторных)

Код зан.	Тема и краткое содержание занятия	Вид зан.	Кол. часов	Компетенции	УМИО
1	2	3	4	5	6
Курс 3, Семестр 6					
Модуль 1: Основные способы построения программных архитектур					
1.1	Лекция №1. Знакомство с DevOps. Основные способы построения программных архитектур. Выбор архитектуры: монолит vs микросервисы. Стратегия миграции.	Лек.	2	ПК-3	Л1.1 Л2.1
1.2	Практическое занятие №1. Знакомство с DevOps	ПЗ	2	ПК-3	Л1.1

					Л2.1
1.3	Лекция №2 Приложения раньше и сейчас. От виртуальных машин к контейнерам.	Лек.	2	ПК-3	Л1.1 Л2.1
1.4	Практическое занятие №2. Особенности контейнерного подхода	ПЗ	2	ПК-3	Л1.1 Л2.1
1.5	Лекция №3. Контейнеры с точки зрения хранения данных. Особенности хранения данных.	Лек.	2	ПК-3	Л1.1 Л2.1
1.6	Практическое занятие №3. Знакомство с Cloud Native Storage (CNS)	ПЗ	2	ПК-3	Л1.1 Л2.1
1.7	Лабораторная работа №1. Работа с Cloud Native Storage (CNS)	ЛР	4	ПК-3	Л1.1 Л2.1
1.8	Лабораторная работа №2. Cloud Native Storage в vSphere и vSAN.	ЛР	4	ПК-3	Л1.1 Л2.1
1.9	Лекция №4. Архитектурные особенности. Приоритеты команд в рамках запуска приложений. Привычные инструменты для разработчика.	Лек.	2	ПК-3	Л1.1 Л2.1
1.10	Практическое занятие №4. Концепция vSphere with Tanzu	ПЗ	2	ПК-3	Л1.1 Л2.1
1.11	Лабораторная работа №3. Использование vSphere with Tanzu	ЛР	4	ПК-3	Л1.1 Л2.1
1.12	Практическое занятие №5. Изучение приоритетов команд в рамках запуска приложений	ПЗ	2	ПК-3	Л1.1 Л2.1
1.13	Лабораторная работа №4. Использование приоритетов команд в рамках запуска приложений.	ЛР	4	ПК-3	Л1.1 Л2.1
1.14	Каскадная модель разработки. Особенности методологии DevOps. Виды архитектур: концепция, плюсы и минусы. Метрики результата. Предпосылки для миграции. Стратегия миграции. Особенности хранения данных. Знакомство с Cloud Native Storage (CNS). Cloud Native Storage в vSphere и vSAN. vSphere with Tanzu: концепция. Приоритеты команд в рамках запуска приложений. vSphere со встроенным Kubernetes. Привычные инструменты для администратора.	СРС	24	ПК-3	Л1.1 Л2.1
Модуль 2: Инструменты автоматизации инфраструктуры					
2.1	Лекция №5. Особенности мониторинга. Мониторинг сети и визуализация. Зоны ответственности DevOps-инженера.	Лек.	2	ПК-3	Л1.1 Л2.1
2.2	Практическое занятие №6. стек технологий для запуска микросервисных приложений.	ПЗ	2	ПК-3	Л1.1 Л2.1
2.3	Лекция №6. Популярные инструменты автоматизации инфраструктуры.	Лек.	2	ПК-3	Л1.1 Л2.1
2.4	Практическое занятие №7. Концепция «Инфраструктура как код»	ПЗ	2	ПК-3	Л1.1 Л2.1
2.5	Практическое занятие №8. Кросс-облачная сеть и безопасность	ПЗ	2	ПК-3	Л1.1 Л2.1
2.6	Лекция №7. Области внимания DevSecOps и Supply Chain Security	Лек.	2	ПК-3	Л1.1 Л2.1
2.7	Лабораторная работа №5. Реализация CI/CD для	ЛР	4	ПК-3	Л1.1

	Azure IoT Edge				Л2.1
2.8	Лабораторная работа №6. Инструменты IoT Edge для CI/CD	ЛР	4	ПК-3	Л1.1 Л2.1
2.9	Лекция №8. Основные понятия концепции MLOps.	Лек.	2	ПК-3	Л1.1 Л2.1
2.10	Лабораторная работа №7. Жизненный цикл разработки ML/AI	ЛР	4	ПК-3	Л1.1 Л2.1
2.11	Лабораторная работа №8. Знакомство с MLOps-платформой	ЛР	4	ПК-3	Л1.1 Л2.1
2.12	Зоны ответственности DevOps-инженера, администратора, разработчика. стек технологий для запуска микросервисных приложений. Внедрение инноваций, внутренняя разработка и другие причины мультиоблачной стратегии. Автоматизация и интеграция. DevSecOps с GitHub Advanced Security. Azure IoT Edge: опыт разработчиков. Инструменты CI/CD для Azure IoT Edge. Инструменты IoT Edge для CI/CD. Основные понятия концепции MLOps. Жизненный цикл разработки ML/AI. Какие задачи решает DevOps/MLOps на разных этапах жизненного цикла. Знакомство с MLOps-платформой	СРС	20	ПК-3	Л1.1 Л2.1
	Экзамен		36		
Итого			144		

4.2. Заочная форма обучения, 5лет (всего 144 часов, из них 22 час аудиторных)

Код зан.	Тема и краткое содержание занятия	Вид зан.	Кол. часов	Компетенции	УМИО
1	2	3	4	5	6
Курс 3, Семестр 6					
Модуль 1: Основы построения специализированных процессоров					
1.1	Лекция №1. Знакомство с DevOps. Основные способы построения программных архитектур. Выбор архитектуры: монолит vs микросервисы. Стратегия миграции.	Лек.	2	ПК-3	Л1.1 Л2.1
1.2	Лекция №2 Контейнеры с точки зрения хранения данных. Особенности хранения данных.	Лек.	2	ПК-3	Л1.1 Л2.1
1.3	Практическое занятие №1. Cloud Native Storage (CNS)	ПЗ	2	ПК-3	Л1.1 Л2.1
1.4	Практическое занятие №2. Cloud Native Storage в vSphere и vSAN.	ПЗ	2	ПК-3	Л1.1 Л2.1
1.5	Каскадная модель разработки. Особенности методологии DevOps. Виды архитектур: концепция, плюсы и минусы. Метрики результата. Предпосылки для миграции. Стратегия миграции. Особенности хранения данных. Знакомство с Cloud Native Storage (CNS). Cloud Native Storage в vSphere и vSAN. vSphere with Tanzu: концепция. Приоритеты команд в рамках запуска приложений. vSphere со встроенным Kubernetes. Привычные ин-	СРС	28	ПК-3	Л1.1 Л2.1

	струменты для администратора.				
Модуль 2: Периферия специализированных процессоров					
2.1	Лекция №3. Особенности мониторинга. Мониторинг сети и визуализация. Зоны ответственности DevOps-инженера. Популярные инструменты автоматизации инфраструктуры.	Лек.	2	ПК-3	Л1.1 Л2.1
2.2	Практическое занятие №3. Концепция «Инфраструктура как код»	ПЗ	2	ПК-3	Л1.1 Л2.1
2.3	Лекция №4. Области внимания DevSecOps и Supply Chain Security. Основные понятия концепции MLOps.	Лек.	2	ПК-3	Л1.1 Л2.1
2.6	Лабораторная работа №1. Реализация CI/CD для Azure IoT Edge	ЛР	2	ПК-3	Л1.1 Л2.1
2.7	Лабораторная работа №2. Инструменты IoT Edge для CI/CD	ЛР	2	ПК-3	Л1.1 Л2.1
2.8	Лабораторная работа №3. Жизненный цикл разработки ML/AI	ЛР	2	ПК-3	Л1.1 Л2.1
2.9	Лабораторная работа №4. Знакомство с MLOps-платформой	ЛР	2	ПК-3	Л1.1 Л2.1
2.10	Зоны ответственности DevOps-инженера, администратора, разработчика. стек технологий для запуска микросервисных приложений. Внедрение инноваций, внутренняя разработка и другие причины мультиоблачной стратегии. Автоматизация и интеграция. DevSecOps с GitHub Advanced Security. Azure IoT Edge: опыт разработчиков. Инструменты CI/CD для Azure IoT Edge. Инструменты IoT Edge для CI/CD.	СРС	85	ПК-3	Л1.1 Л2.1
	Экзамен		9		
Итого			144		

5. Учебно-методическое и информационное обеспечение дисциплины

5.1 Рекомендуемая литература				
5.1.1. Основная литература				
Код	Авторы, составители	Заглавие	Издательство, год	Кол.
Л1.1	Баланов А. Н.	DevOps: интеграция и автоматизация: Учебное пособие для вузов	Санкт-Петербург: Лань, 2025	Э1
5.1.2 Дополнительная литература				
Код	Авторы, составители	Заглавие	Издательство, год	Кол.
Л2.1	Федосеева Т.А.	Автоматизированные технологии управления проектами: учебно-методическое пособие	Москва: МИСИ-МГСУ, ЭБС АСВ, 2021.	Э2
5.1.3 Методическое обеспечение для самостоятельной работы обучающихся				
Код	Авторы, составители	Заглавие	Издательство, год	Кол.
Л3.1	Сейерс Э. Х. , Милл А.	Docker на практике	М.: ДМК Пресс, 2020	Э3

5.2 Электронные образовательные ресурсы	
Э1	https://e.lanbook.com/book/440162
Э2	https://e.lanbook.com/book/262268
Э3	https://e.lanbook.com/book/131719
5.3 Программное обеспечение	
П.1	LibreOffice - свободное ПО
П.2	Oracle VirtualBox

6. Материально-техническое обеспечение дисциплины

6.1 МТО лекционных занятий	
1	Лекционная аудитория, оснащенная проектором, ПК (ноутбуком), экраном.
6.2 МТО лабораторных работ и практических занятий	
1	Учебные аудитории, оборудованные компьютерной техникой, с возможностью подключения к сети «Интернет», и обеспечением доступа в электронную информационно-образовательную среду СКФ МТУСИ
6.3 МТО рубежных контролей и зачетов	
1	Учебные аудитории, оборудованные компьютерной техникой с возможностью выхода в локальную сеть Филиала и Интернет.
6.4 МТО самостоятельной работы обучающихся	
1	Помещение для самостоятельной работы обучающихся оснащено компьютерной техникой с возможностью подключения к сети Интернет и обеспечением доступа в электронную информационно-образовательную среду СКФ МТУСИ

7. Оценочные материалы
Оценочные материалы и перечень видов оценочных средств текущего контроля и промежуточной аттестации представлены в Приложении 1 к рабочей программе дисциплины.

8. Методические рекомендации для обучающихся по освоению дисциплины
Методические рекомендации по освоению дисциплины представлены в Приложении 2 к рабочей программе дисциплины.

9. 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).

10. Дополнения и изменения в Рабочей программе

Лист актуализации рабочей программы дисциплины «Основы DevOps Б1.В.05» для использования в 20___/20___ учебном году

Утверждаю

Зам. директора по УР _____
«___» _____ 20___ г.

Направление: **09.03.01. Информатика и вычислительная техника**

Профиль: **Искусственный интеллект и машинное обучение**

Форма обучения: очная, заочная

(Возможны следующие варианты):

- а) Рабочая программа действует без изменений.
- б) В рабочую программу вносятся следующие изменения:
 - 1)
 - 2)
 - 3)

Разработчик (и): _____
(ФИО, ученая степень, ученое звание)
«___» _____ 20___ г.

Рабочая программа пересмотрена и одобрена на заседании кафедры ИВТ
Протокол № _____ от «___» _____ 20___ г.

Заведующий кафедрой _____

Оценочные материалы для проведения текущего контроля и промежуточной аттестации по дисциплине

1. Оценочные материалы для проведения промежуточной аттестации по дисциплине

1.1 Шкала оценивания компетенций

Шкала оценивания компетенций		
Оценка	Уровень освоения компетенции	Критерии оценивания
«Отлично»	Высокий уровень	Обучающийся показывает всестороннее, систематическое и глубокое знание основного и дополнительного учебного материала, дает полные, профессиональные и грамотные ответы на поставленные вопросы, что свидетельствует об освоении обучающимся взаимосвязи основных понятий дисциплины; усвоил основную и знаком с дополнительной рекомендованной литературой; проявляет творческие способности в понимании, изложении и использовании учебно-программного материала.
«Хорошо»	Повышенный уровень	Обучающийся показывает достаточный уровень знания основного и дополнительного учебного материала, дает в целом грамотные ответы на поставленные вопросы, что свидетельствует об освоении обучающимся взаимосвязи основных понятий дисциплины; усвоил основную и знаком с дополнительной рекомендованной литературой. Допускает не существенные погрешности в ответах, устраняет их без помощи преподавателя.
«Удовлетворительно»	Пороговый уровень	Обучающийся демонстрирует знание основного учебного материала в минимальном объеме, необходимом для дальнейшей учебы; в целом отвечает на поставленные вопросы, допуская не принципиальные ошибки; знаком с основной литературой. Допускает ошибки в ответах различного уровня, которые исправляет после наводящих вопросов и под руководством преподавателя.
«Неудовлетворительно»	Минимальный уровень не достигнут	Обучающийся обнаруживает существенные пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении, предусмотренных программой, заданий; не знаком с рекомендованной литературой, не может исправить допущенные ошибки.

		Как правило, оценка «неудовлетворительно» ставится обучающимся, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.
Экзамен		0-100 «Неудовлетворительно» - 0-40 «Удовлетворительно» - 41-60 «Хорошо» - 61-80 «Отлично» - 81-100

1.2 Показатели, критерии и шкалы оценивания компетенций

Показатели компетенции	Критерии оценивания	Шкала оценивания
ПК-3: Способен разрабатывать и внедрять программное обеспечение, осуществлять его поддержку		
Знать:		
– методы разработки, внедрения и сопровождения программного обеспечения	Тестовый контроль	Модуль 1 0÷20 «Неудовлетворительно» - 0-8 «Удовлетворительно» - 9-13 «Хорошо» - 14-18 «Отлично» -19-20
	Тестовый контроль	Модуль 2 0÷20 «Неудовлетворительно» - 0-8 «Удовлетворительно» - 9-13 «Хорошо» - 14-18 «Отлично» -19-20
Уметь:		
– разрабатывать программное обеспечение с применением современных средств и технологий в соответствии с требованиями;	Практические занятия 2,3	Модуль 1 0÷4 за каждое ПЗ «Неудовлетворительно» - 0-1 «Удовлетворительно» - 2 «Хорошо» - 3 «Отлично» - 4
– выполнять внедрение и сопровождение программного обеспечения;	Лабораторные работы 1,2,3	0÷6 за каждую ЛР «Неудовлетворительно» - 0-1 «Удовлетворительно» - 2 «Хорошо» - 3 -4 «Отлично» -5-6
– выбирать эффективные алгоритмы решения задач профессиональной деятельности.		
Владеть:		
– приемами эффективного решения задач в рамках профессиональной деятельности	Практическое занятие 1	Модуль 1 0÷4 «Неудовлетворительно» - 0-1 «Удовлетворительно» - 2 «Хорошо» - 3 «Отлично» -4

	Практическое занятие 4,5	Модуль 2 0÷6 за каждое ПЗ «Неудовлетворительно» - 0-1 «Удовлетворительно» - 2 «Хорошо» - 3-4 «Отлично» - 5-6
	Лабораторные работы 4,5,6	0÷6 за каждую ЛР «Неудовлетворительно» - 0-1 «Удовлетворительно» - 2 «Хорошо» - 3-4 «Отлично» - 5-6
	Экзамен	0-100 «Неудовлетворительно» - 0-40 «Удовлетворительно» - 41-60 «Хорошо» - 61-80 «Отлично» - 81-100

1.3 Оценочные материалы: типовые контрольные задания, иные материалы

1.3.1 Оценочные материалы текущего контроля для очной формы обучения

Данная дисциплина формирует у обучающегося 1 компетенцию, которая может быть оценена при ответе обучающегося на контрольный вопрос или при выполнении практического задания.

Модуль 1 (50 баллов)

Модуль содержит 4 лекционных, 5 практических и 4 лабораторных занятий. Знание лекционного материала оценивается с помощью тестового контроля, максимальное количество баллов за тестовый контроль составляет 20. За практические и лабораторные занятия обучающийся получает максимум 30 баллов. Общее максимальное количество баллов за модуль 1 составляет 50.

Практическое занятие №1. (ПК-3)

Контрольные вопросы к ПЗ.

1. Что такое Linux?
2. Что происходит в подробностях, когда вы вводите `cat file.txt` ?
3. Что такое файловый дескриптор? Какие создаются по умолчанию?
4. Нужно ли делать резервные копии? А что еще нужно делать?
5. Как поставить пакет? Что такое apt?

Лабораторная работа №1. (ПК-3)

Контрольные вопросы к ЛР.

1. Как определяются права на файл? Как сделать файл исполняемым?
2. Как создать пользователя в GNU/Linux?
3. Как пользоваться `man`?
4. Как выдать права на `sudo`?
5. Зачем `.bashrc` файл?

Практическое занятие №2. (ПК-3)

Контрольные вопросы к ПЗ.

1. Как посмотреть занятое место на диске?
2. Как подключиться к серверу по ssh? Подробно о всех возможных проблемах.
3. Что такое публичный и приватный ключ? Какой оставляем себе, а какой копируем на сервер?
4. ssh-agent - зачем?
5. В чем разница между soft и hard лимитами?
6. Кто такой суперпользователь и что ему можно?

Лабораторная работа №2. (ПК-3)

Контрольные вопросы к ЛР.

1. Что такое и зачем нужен Docker? Альтернативные системы?
2. Как получить Docker-образ, что это такое?
3. Как запустить контейнер? Как получить доступ к его портам?
4. Как просмотреть логи контейнера?
5. Как сохранить данные внутри контейнера между его перезапусками?

Лабораторная работа №3. (ПК-3)

Контрольные вопросы к ЛР.

1. Как подключить контейнеры к одной сети? Какие есть альтернативные варианты?
2. Почему контейнеры могут обращаться между собой по имени (хэшу, если его нет)?
3. Что такое метки (docker tag)?
4. Как удалить ненужные образа и контейнеры?

Практическое занятие №3. (ПК-3)

Контрольные вопросы к ПЗ.

10. Как запустить что-то внутри работающего контейнера?
11. Как узнать, какие файлы изменяет программа внутри контейнера?
12. Когда происходит завершение контейнера? Как сделать?
13. Перезапустите сборку собранного образа, оцените время пересборки, объясните причины.

Модуль 2 (50 баллов).

Модуль содержит 4 лекционных, 3 практических и 4 лабораторных занятий. Знание лекционного материала оценивается с помощью тестового контроля, максимальное количество баллов за тестовый контроль составляет 20. За практические и лабораторные занятия обучающийся получает максимум 30 баллов. Общее максимальное количество баллов за модуль 1 составляет 50.

Лабораторная работа №4. (ПК-3)

Контрольные вопросы к ЛР.

1. К какому числу слоев стремиться в образе, правила оптимизации?
2. Опишите базовые команды Dockerfile, что они делают, где смотреть документацию?
3. Что такое контекст сборки, как его оптимизировать?

Практическое занятие №4. (ПК-3)

Контрольные вопросы к ПЗ.

1. Что входит в состав ansible playbook?
2. Что такое группы хостов?
3. Как одной командой "пингануть" группу хостов?

Лабораторная работа №5. (ПК-3)

Контрольные вопросы к ЛР.

1. Что такое kubernetes и зачем он нужен?
2. Расскажите про основные компоненты kubernetes и покажите их в kube-systems
3. Расскажите про создание пода из yaml, основные поля.
4. Расскажите про сервисы ClusterIp и Headless
5. Расскажите про сервисы NodePort и LoadBalancer

Лабораторная работа №6. (ПК-3)

Контрольные вопросы к ЛР.

1. Расскажите про Ingress
2. Расскажите про Deployment, StatefulSet и DaemonSet
3. Расскажите про Job и CronJob
4. Расскажите про ConfigMap и Secret
5. Перечислите основные команды kubectl, рассмотренные в лабе

Практическое занятие №5. (ПК-3)

Контрольные вопросы к ПЗ.

1. Что означает запустить код в гитлаб?
2. Как настроить CI?
3. Описать Dockerfile и собрать образ.
4. Как создать configmap, service, deployment и ingress в кубере?
5. Перечислить настройки CI/CD для автоматического обновления.

1.3.2. Вопросы и задания, выносимые на экзамен по дисциплине «Основы DevOps» (ПК-3) (промежуточная аттестация):

Вопросы:

1. Как подключиться к серверу по ssh? Основные принципы работы с публичным и приватным ключами. Принципы работы с ssh-agent.
2. Система прав в GNU/Linux.
3. Системные вызовы в GNU/Linux, userspace/kernelspace.
4. Пользователи в GNU/Linux, суперпользователь, su и sudo.
5. Системы управления пакетами в GNU/Linux.
6. Системы виртуализации. Гипервизоры 1 и 2 типа.
7. Что такое Docker и для чего он используется? Какие существуют альтернативы Docker?
8. Docker образ и контейнер. Что такое метки (docker tag)? Запуск контейнера, остановка, просмотр логов.
9. Как сохранить данные внутри контейнера между его перезапусками?
10. Как подключить контейнеры к одной сети? Какими способами можно решить данную задачу? Как получить доступ к порту приложения внутри контейнера?
11. Почему контейнеры могут общаться между собой по имени (или хэшу, если нет имени)?

12. Как запустить код внутри работающего контейнера? Как узнать, какие файлы изменены внутри контейнера?
13. Сборка образов в Docker. К какому количеству слоев нужно стремиться в образе? Каковы правила оптимизации при формировании образа?
14. Опишите структуру docker-compose файла. Зачем нужен docker-compose?
15. Методы балансировки нагрузки и рекомендации по их использованию.
16. Пирамида автоматизации тестирования. Семь принципов тестирования (в соответствии с ISTQB).
17. Что такое GitFlow и GitHub Flow. Преимущества и недостатки каждой модели?
18. Типичные этапы итерации процесса разработки с применением CI/CD.
19. Управление исходным кодом. Виды систем контроля версий.
20. Как осуществляется мониторинг, какие существуют типы мониторинга?
21. Мониторинг с помощью Prometheus-стека.
22. Как импортировать готовый dashboard Grafana – возможно ли добавление в него новой панели? Какие источники поддерживает Grafana?
23. Что такое облако, VPS/VDS? В чем разница между публичным и приватным облаком? Основные объекты, предоставляемые облаками. IaaS, PaaS, SaaS.
24. OpenStack. Назначение и функциональность. В каких случаях компаниям целесообразно использовать OpenStack в качестве компонента IT-инфраструктуры? Чем можно заменить OpenStack?
25. Что такое кластер? Опишите кластер на примере Kubernetes.
26. Горизонтальное и вертикальное масштабирование.
27. Основные типы ресурсов в Kubernetes. Основные компоненты кластера Kubernetes и их назначение.
28. Принципы работы с Pod и Volume в Kubernetes. DaemonSet, StatefulSet.
29. Deployment. Service. Стратегии обновления приложения. Основные принципы работы с сервисами в Kubernetes.
30. Основные принципы работы с ConfigMap и Secret. В чем различие между ними?

1.4 Оценочные материалы для заочной формы обучения

Практическое занятие №1. (ПК-3)

Контрольные вопросы к ПЗ.

1. Что такое Linux?
2. Что происходит в подробностях, когда вы вводите cat file.txt ?
3. Что такое файловый дескриптор? Какие создаются по умолчанию?
4. Нужно ли делать резервные копии? А что еще нужно делать?
5. Как поставить пакет? Что такое apt?

Лабораторная работа №1. (ПК-3)

Контрольные вопросы к ЛР.

1. Как определяются права на файл? Как сделать файл исполняемым?
2. Как создать пользователя в GNU/Linux?
3. Как пользоваться man?
4. Как выдать права на sudo?
5. Зачем .bashrc файл?

Практическое занятие №2. (ПК-3)

Контрольные вопросы к ПЗ.

1. Как посмотреть занятое место на диске?
2. Как подключиться к серверу по ssh? Подробно о всех возможных проблемах.
3. Что такое публичный и приватный ключ? Какой оставляем себе, а какой копируем на сервер?

4. ssh-agent - зачем?
5. В чем разница между soft и hard лимитами?
6. Кто такой суперпользователь и что ему можно?

Лабораторная работа №2. (ПК-3)

Контрольные вопросы к ЛР.

1. Что такое и зачем нужен Docker? Альтернативные системы?
2. Как получить Docker-образ, что это такое?
3. Как запустить контейнер? Как получить доступ к его портам?
4. Как просмотреть логи контейнера?
5. Как сохранить данные внутри контейнера между его перезапусками?

Лабораторная работа №3. (ПК-3)

Контрольные вопросы к ЛР.

1. Как подключить контейнеры к одной сети? Какие есть альтернативные варианты?
2. Почему контейнеры могут обращаться между собой по имени (хэшу, если его нет)?
3. Что такое метки (docker tag)?
4. Как удалить ненужные образы и контейнеры?

Практическое занятие №3. (ПК-3)

Контрольные вопросы к ПЗ.

10. Как запустить что-то внутри работающего контейнера?
11. Как узнать, какие файлы изменяет программа внутри контейнера?
12. Когда происходит завершение контейнера? Как сделать?
13. Перезапустите сборку собранного образа, оцените время пересборки, объясните причины.

Лабораторная работа №4. (ПК-3)

Контрольные вопросы к ЛР.

1. К какому числу слоев стремиться в образе, правила оптимизации?
2. Опишите базовые команды Dockerfile, что они делают, где смотреть документацию?
3. Что такое контекст сборки, как его оптимизировать?

Практическое занятие №4. (ПК-3)

Контрольные вопросы к ПЗ.

4. Что входит в состав ansible playbook?
5. Что такое группы хостов?
6. Как одной командой "пингануть" группу хостов?

Лабораторная работа №5. (ПК-3)

Контрольные вопросы к ЛР.

6. Что такое kubernetes и зачем он нужен?
7. Расскажите про основные компоненты kubernetes и покажите их в kube-systems
8. Расскажите про создание пода из yaml, основные поля.
9. Расскажите про сервисы ClusterIp и Headless
10. Расскажите про сервисы NodePort и LoadBalancer

Лабораторная работа №6. (ПК-3)

Контрольные вопросы к ЛР.

6. Расскажите про Ingress
7. Расскажите про Deployment, StatefulSet и DaemonSet
8. Расскажите про Job и CronJob
9. Расскажите про ConfigMap и Secret
10. Перечислите основные команды kubectl, рассмотренные в лабе

Практическое занятие №5. (ПК-3)

Контрольные вопросы к ПЗ.

6. Что означает запустить код в гитлаб?
7. Как настроить CI?
8. Описать Dockerfile и собрать образ.
9. Как создать configmap, service, deployment и ingress в кубере?
10. Перечислить настройки CI/CD для автоматического обновления.

Экзаменационные вопросы такие же, как и для студентов очной формы обучения

2. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций

2.1. Порядок и методика проведения текущего контроля и промежуточной аттестации.

Предусмотрены следующие виды контроля:

- текущий контроль по каждому модулю – в форме выполнения тестового контроля для оценки теоретических знаний;
- текущий контроль по каждому модулю – в форме отчетов по практическим занятиям для оценки практических навыков;
- промежуточная аттестация по дисциплине – в форме экзамена.

Текущий контроль успеваемости в форме защиты лабораторных и практических занятий проводится в два этапа:

- 1-й этап: допуск к выполнению лабораторной работы – проводится в форме письменной «летучки» (5-10 мин) с целью контроля знаний обучающимися теоретической части лабораторной работы и готовности к выполнению практических исследований;

- 2-й этап выполняется по окончании каждой лабораторной работы (практического занятия) в форме индивидуального собеседования по выполненным исследованиям или расчетам. Проводится с целью контроля закрепления теоретической части материала и степени отработки обучающимся практических навыков исследования на аппаратуре.

С целью повышения качества обучения за счет побуждения обучающихся к активной текущей учебной работе, четкого и оперативного контроля всего хода учебного процесса, снижения роли случайных и субъективных факторов при оценивании учебной деятельности обучающихся в образовательном процессе реализована модульно-рейтинговая система.

Правила ее использования описаны в «Положении об МРС».

Набранные обучающимися баллы могут быть переведены в оценку:

- «неудовлетворительно» - от 0% до 40% от максимального количества баллов;
- «удовлетворительно» - от 41% до 60% максимального количества баллов;
- «хорошо» - от 61% до 80% максимального количества баллов;
- «отлично» - от 81% до 100% максимального количества баллов.

Для получения экзамена обучающемуся достаточно набрать 41 и более баллов.

Соотношения максимального количества баллов, полученных обучающимся по блокам модулей, показаны в Таблице.

Таблица - Распределение баллов по блокам модулей дисциплины «Основы DevOps»

Модуль	Всего баллов (Максимальное значение)	Теоретический блок (тестирование)	Практический блок
Модуль 1	50	20	30
Модуль 2	50	20	30
Экзамен			100

На экзамене производится оценка тех компетенций, которые должны быть в той или иной форме освоены в процессе изучения. Рекомендуется формировать вопросы в билетах таким образом, чтобы преподаватель смог оценить все компетенции данной дисциплины.

3. Промежуточная аттестация

Промежуточная аттестация проводится в форме экзамена. Экзамен проводится по расписанию в устном виде по экзаменационным билетам. Результаты аттестации заносятся в ведомость и зачетную книжку обучающегося.

Образец экзаменационного билета

	МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ Северо-Кавказский филиал ордена Трудового Красного Знамени федерального государственного бюджетного образовательного учреждения высшего образования «Московский технический университет связи и информатики»	Утверждаю Зав. кафедрой «ИВТ» _____ Соколов С.В. «___» _____ 20__ г.
Направление подготовки: 09.03.01. Информатика и вычислительная техника Курс 3 Дисциплина: Основы DevOps		
Билет №1 1. Что такое Docker и для чего он используется? Какие существуют альтернативы Docker? 2. Основные типы ресурсов в Kubernetes. Основные компоненты кластера Kubernetes и их назначение. Ст. преподаватель каф. «ИВТ» _____ Коноваленко А.В. «___» _____ 20__ г.		

Один комплект отпечатанных экзаменационных билетов, подписанных преподавателем кафедры и утвержденных заведующим кафедрой хранится у заведующего кафедрой, другой комплект – у преподавателя, ведущего дисциплину.

4 Тестовые задания для проведения оценки сформированности компетенций (ПК-3)

Один комплект тестовых вопросов с указанием правильных ответов хранится у заведующего кафедрой, другой комплект – у преподавателя, ведущего дисциплину.

Тестовые задания позволяют оценить уровень сформированности компетенции ПК-3.

Задания распределены по блокам в соответствии с уровнем сложности. Каждый блок содержит номер задания и текст задания.

Базовый уровень содержит примерно 50 % заданий.

Он формируется из заданий с выбором одного или нескольких верных ответов из предложенных или воспроизведения фактического материала (терминология, факты, классификация, параметры и др.).

Повышенный уровень в среднем составляет 35 % заданий.

В нем используются задания на сопоставление, сравнение, установление последовательности;

Высокий уровень в среднем составляют – 15 % заданий.

В заданиях используются: решения нетиповых задач, алгоритмы, доказательства, задания с развернутым ответом – определения, перечисление, изображение схемы, структуры и др.

Количество заданий равно 20.

ПК-3: Способен разрабатывать и внедрять программное обеспечение, осуществлять его поддержку

БЛОК А (базовый уровень) – Задание с выбором одного или нескольких верных ответов из предложенных

Инструкция по тестам Блока А: Прочитайте текст и выберите один или несколько правильных ответов

№ задания	Тексты заданий
1	Что такое Linux? 1. семейство Linux –подобных операционных систем 2. семейство Unix-подобных операционных систем на базе ядра Linux. 3. семейство Linux –подобных операционных систем на базе ядра Unix
2	Что такое файловый дескриптор? 1. это уникальный числовой идентификатор 2. ссылка 3. параметр запуска
3	При помощи команды APT мы можем: 1. устанавливать программные пакеты 2. обновлять программные пакеты 3. удалять программные пакеты 4. всё выше перечисленное
4	Команда Man служит для: 1. вызова справочной страницы по командам 2. вызова помощника 3. вызова справочной страницы по командам, функциям или программам 4. вызова справочной страницы по функциям
5	Команда Sudo служит для: 1. запуска программы с привилегиями другой учётной записи 2. контроля доступа

№ задания	Тексты заданий
	3. аудита и протоколирования команд 4. все вышеперечисленное
6	Суперпользователь это: 1. пользователь со сверхспособностями 2. системный администратор с максимальными привилегиями 3. пользователь наделенных суперспособностями 4. пользователь с правами администратора
7	Docker это: 1. работник дока 2. программное обеспечение для разработки проектов 3. программное обеспечение с открытым исходным кодом для автоматизации развёртывания и управления приложениями в средах с поддержкой контейнеризации. 4. программное обеспечение для автоматической разработки приложений
8	Kubernetes это: 1. функция в программировании 2. вспомогательный алгоритм 3. автоматическое развёртывание приложений 4. платформа для оркестрации контейнеров в кластере
9	CI/CD это: 1. это методология автоматизации тестирования и доставки новых модулей разрабатываемого проекта заинтересованным сторонам 2. это методология автоматизации тестирования новых модулей разрабатываемого проекта заинтересованным сторонам 3. это методология автоматизации доставки новых модулей разрабатываемого проекта заинтересованным сторонам 4. нет правильного ответа
10	GitLab это: 1. вычислительный центр 2. платформа для управления проектами и репозиториями программного кода 3. лабораторный комплекс 4. платформа для доступа к облачным сервисам

БЛОК Б (повышенный уровень) – Задание закрытого типа на установление соответствия

Инструкция: Прочитайте текст и установите соответствие или последовательность

№ задания	Тексты заданий
11	Выстройте в правильном порядке этапы настройки CI/CD: 1. тестирование 2. сборка 3. мониторинг 4. развёртывание

№ задания	Тексты заданий								
12	Найдите соответствие базовых команд и действий Dockerfile: <table> <tr> <td>1) FROM</td><td>А) Копирует файлы и директории из контекста сборки в контейнер.</td></tr> <tr> <td>2) RUN</td><td>Б) Определяет команду по умолчанию, которую нужно выполнить при запуске контейнера</td></tr> <tr> <td>3) COPY</td><td>В) Задаёт базовый (родительский) образ для нового образа Docker.</td></tr> <tr> <td>4) CMD</td><td>Г) Создаёт слой образа в процессе сборки образа.</td></tr> </table>	1) FROM	А) Копирует файлы и директории из контекста сборки в контейнер.	2) RUN	Б) Определяет команду по умолчанию, которую нужно выполнить при запуске контейнера	3) COPY	В) Задаёт базовый (родительский) образ для нового образа Docker.	4) CMD	Г) Создаёт слой образа в процессе сборки образа.
1) FROM	А) Копирует файлы и директории из контекста сборки в контейнер.								
2) RUN	Б) Определяет команду по умолчанию, которую нужно выполнить при запуске контейнера								
3) COPY	В) Задаёт базовый (родительский) образ для нового образа Docker.								
4) CMD	Г) Создаёт слой образа в процессе сборки образа.								
13	Соберите правильную последовательность команды создания нового пользователя в GNU/Linux с домашним каталогом и оболочкой: <table> <tr> <td>1. -m</td><td>3. -d</td></tr> <tr> <td>2. sudo</td><td>4. useradd</td></tr> </table>	1. -m	3. -d	2. sudo	4. useradd				
1. -m	3. -d								
2. sudo	4. useradd								
14	Определите правильный синтаксис команды apt: <пакет(ы)> <действие> - sudo <доп. параметры> - apt								
15	Определите порядок управления жизненным циклом контейнерных приложений: - Масштабирование - Управление ресурсами - Развертывание - Обновление - Мониторинг								
16	Найдите соответствие определений: <table> <tr> <td>1. Docker</td><td>А. Методология автоматизации процессов разработки программного обеспечения</td></tr> <tr> <td>2. CI/CD</td><td>Б. Программная платформа для автоматического управления контейнеризованными приложениями.</td></tr> <tr> <td>3. Kubernetes</td><td>В) платформа контейнеризации с открытым исходным кодом для разработки, тестирования, доставки и запуска приложений в средах с поддержкой контейнеризации</td></tr> </table>	1. Docker	А. Методология автоматизации процессов разработки программного обеспечения	2. CI/CD	Б. Программная платформа для автоматического управления контейнеризованными приложениями.	3. Kubernetes	В) платформа контейнеризации с открытым исходным кодом для разработки, тестирования, доставки и запуска приложений в средах с поддержкой контейнеризации		
1. Docker	А. Методология автоматизации процессов разработки программного обеспечения								
2. CI/CD	Б. Программная платформа для автоматического управления контейнеризованными приложениями.								
3. Kubernetes	В) платформа контейнеризации с открытым исходным кодом для разработки, тестирования, доставки и запуска приложений в средах с поддержкой контейнеризации								
17	Найдите соответствия компонентов Docker: <table> <tr> <td>1. Docker daemon</td><td>А) Неизменяемый образ, из которого разворачивается контейнер.</td></tr> <tr> <td>2. Docker client</td><td>Б) Фоновый процесс, который управляет контейнерами, образами и другими ресурсами.</td></tr> <tr> <td>3. Docker image</td><td>В) Развёрнутое и запущенное приложение.</td></tr> <tr> <td>4. Docker container</td><td>Г) Интерфейс командной строки, через который пользователи взаимодействуют с демоном и отправляют ему команды.</td></tr> </table>	1. Docker daemon	А) Неизменяемый образ, из которого разворачивается контейнер.	2. Docker client	Б) Фоновый процесс, который управляет контейнерами, образами и другими ресурсами.	3. Docker image	В) Развёрнутое и запущенное приложение.	4. Docker container	Г) Интерфейс командной строки, через который пользователи взаимодействуют с демоном и отправляют ему команды.
1. Docker daemon	А) Неизменяемый образ, из которого разворачивается контейнер.								
2. Docker client	Б) Фоновый процесс, который управляет контейнерами, образами и другими ресурсами.								
3. Docker image	В) Развёрнутое и запущенное приложение.								
4. Docker container	Г) Интерфейс командной строки, через который пользователи взаимодействуют с демоном и отправляют ему команды.								

БЛОК В (высокий уровень) – Задание открытого типа с развернутым ответом

Инструкция: Прочитайте текст и запишите обоснованный ответ

№ задания	Тексты заданий
18	Процесс развёртывания проекта на сервере или другом рабочем окружении для его запуска и предоставления пользователям.
19	Непрерывная интеграция в разработке программного обеспечения.
20	Платформа для управления проектами и репозиториями, работающая с системой контроля версий Git.

Система оценивания выполнения тестовых заданий по компетенции ОПК-1

Номер задания	Указания по оцениванию	Результат оценивания (баллы, полученные за выполнение задания/характеристика правильности ответа)
Задания 1-10	Задания с выбором одного или нескольких верных ответов из предложенных считается верным, если правильно указана цифра или цифры, означающие верные ответы.	Совпадение с верным ответом оценивается 1 баллом; неверный ответ или его отсутствие - 0 баллов.
Задания 11-17	Задания закрытого типа на установление соответствия считается верным, если правильно установлены все соответствия (позиции из одного столбца верно сопоставлены с позициями другого)	Полное совпадение с верным ответом оценивается 1 баллом; неверный ответ или его отсутствие - 0 баллов.
Задания 18-20	Задание открытого типа с развернутым ответом считается верным, если ответ совпадает с эталонным по содержанию и полноте.	Полный правильный ответ на задание оценивается 3 баллами; если допущена одна ошибка/неточность/ответ правильный, но не полный - 1 балл, если допущено более одной ошибки/ответ неправильный/ответ отсутствует- 0 баллов.

Методика проведения тестирования

Максимальное время выполнения теста - 60 минут.

Количество тестовых заданий в тесте не менее 20.

Для проведения контроля остаточных знаний компьютер формирует в случайном порядке 20 тестовых заданий.

При прохождении тестового контроля знаний студент должен выбрать один или несколько правильных ответов в каждом тестовом задании.

В случае правильного ответа на 16 и более тестовых заданий (81-100% правильных ответов) студент подтверждает освоение компетенций с уровнем «отлично». При ответе на 12-15 тестовых заданий (61-80% правильных ответов) студент подтверждает освоение компетенций с уровнем «хорошо». При ответе на 8-14 тестовых заданий (41-60% правильных ответов) студент подтверждает освоение компетенций с уровнем «удовлетворительно». При ответе менее чем на 8 тестовых заданий (0-40% правильных ответов) студент не подтверждает необходимый уровень знаний и оценивается «неудовлетворительно».

Методические рекомендации по освоению дисциплины

1. Методические рекомендации преподавателю

Дисциплина «**Основы DevOps**» включает в себя:

- лекционные занятия – 16 часов;
- практические занятия – 16 часов;
- лабораторные работы – 32 часа;
- промежуточную аттестацию – экзамен.

Перед началом изучения дисциплины преподаватель должен ознакомить обучающихся с рабочей программой и оценочными материалами по дисциплине, с видами учебной и самостоятельной работы, перечнем литературы и интернет-ресурсов, с формами текущей и промежуточной аттестации, с критериями оценки качества знаний для итоговой оценки по дисциплине.

При проведении лекций преподаватель:

- 1) формулирует тему и цель занятия, объявляет учебные вопросы;
- 2) излагает основные теоретические положения;
- 3) с помощью технических средств обучения и/или под запись дает определения основных понятий, расчетных формул;
- 4) проводит примеры из отечественного и зарубежного опыта, дает текущие статистические данные для наглядного и образного представления изучаемого материала;
- 5) в конце занятия выдает вопросы для самостоятельного изучения.

На занятиях лабораторного цикла следует обратить внимание на соответствие выбираемых обучающимся средств выполнения решаемым в работе задачам.

Каждая лабораторная работа должна быть оформлена и защищена в соответствии с требованиями. Защита производится после оформления отчета по работе.

Во время выполнения заданий в учебной аудитории обучающийся может консультироваться с преподавателем, определять наиболее эффективные методы решения поставленных задач. Если какая-то часть задания остается не выполненной, обучающийся может продолжить её выполнение во время внеаудиторной самостоятельной работы.

Для оценки полученных знаний и освоения учебного материала по каждому разделу и в целом по дисциплине преподаватель использует формы текущего, промежуточного и итогового контроля знаний обучающихся.

2. Методические рекомендации обучающимся

Приступая к изучению новой учебной дисциплины, обучающиеся должны ознакомиться с рабочей программой, оценочными материалами, учебной, научной и методической литературой, имеющейся в ЭИОС СКФ МТУСИ.

Глубина усвоения дисциплины зависит от активной и систематической работы обучающегося на лекциях и практических занятиях, а также в ходе самостоятельной работы, по изучению рекомендованной литературы.

2.1 Методические указания для обучающихся по лекционным занятиям

Важно сосредоточить внимание на содержании лекции. Это поможет лучше воспринимать учебный материал и уяснить взаимосвязь проблем по всей дисциплине.

Основное содержание лекции целесообразнее записывать в тетради в виде ключевых фраз, понятий, тезисов, обобщений, схем, опорных выводов. Необходимо обращать внимание на термины, формулировки, раскрывающие содержание тех или иных явлений и процессов, научные выводы и практические рекомендации. Желательно оставлять в конспектах поля, на которых делать пометки из

рекомендованной литературы, дополняющей материал прослушанной лекции, а также подчеркивающие особую важность тех или иных теоретических положений.

С целью уяснения теоретических положений, разрешения спорных ситуаций необходимо задавать преподавателю уточняющие вопросы. Для закрепления содержания лекции в памяти, необходимо во время самостоятельной работы внимательно прочесть свой конспект и дополнить его записями из учебников и рекомендованной литературы.

Конспектирование читаемых лекций и их последующая доработка способствует более глубокому усвоению знаний, и поэтому являются важной формой учебной деятельности обучающихся.

2.2 Методические указания для обучающихся по подготовке к лабораторным работам

Лабораторные занятия по любой учебной дисциплине проводятся в соответствии с рабочей программой этой учебной дисциплины в установленные расписанием часы.

Инструктаж по технике безопасности, правилам пожарной безопасности и правилам внутреннего распорядка в лаборатории проводится преподавателем на первом лабораторном занятии. На этом же занятии обучающимся сообщаются:

- программа всего предстоящего лабораторного цикла;
- условия взаимодействия обучающихся с преподавателем в процессе выполнения лабораторных работ;
- условия контроля самостоятельной работы обучающихся, включая правила оформления отчетов по лабораторным работам и их последующей защиты;
- другая необходимая информация.

Организация лабораторных работ включает:

- самостоятельную внеаудиторную подготовку обучающегося к выполнению каждой отдельной лабораторной работы в соответствии с ее программой в рамках часов, выделенных на самостоятельную проработку материала в программе дисциплины;
- входной контроль преподавателем степени подготовленности каждого обучающегося к выполнению лабораторных работ;
- выполнение программы лабораторных работ в полном объеме;
- оформление отчета и его защиту каждым обучающимся в установленные сроки;
- формирование преподавателем рейтингов каждого из обучающихся по результатам выполнения и защиты им отдельных лабораторных работ и их циклов (используется Модульно-рейтинговая система).

Оценка качества выполнения лабораторных работ каждым обучающимся производится преподавателем отдельно за подготовку к работе, ее выполнение и защиту.

2.3 Методические указания для обучающихся по подготовке к практическим занятиям

Целью практических занятий является закрепление теоретических знаний, полученных при изучении дисциплины.

При подготовке к практическому занятию целесообразно выполнить следующие рекомендации:

- изучить основную литературу; ознакомиться с дополнительной литературой, новыми публикациями в периодических изданиях: журналах, газетах и т.д.;
- при необходимости доработать конспект лекций. При этом учесть рекомендации преподавателя и требования учебной программы.

При выполнении практических занятий основным методом обучения является самостоятельная работа обучающегося под управлением преподавателя. На них пополняются теоретические знания обучающихся, их умение творчески мыслить, анализировать, обобщать изученный материал, проверяется отношение обучающихся к будущей профессиональной деятельности.

Оценка выполненной работы осуществляется преподавателем комплексно: по результатам выполнения заданий, устному сообщению и оформлению работы.

После подведения итогов занятия обучающийся обязан устранить недостатки, отмеченные преподавателем при оценке его работы.

3. Методические рекомендации для обучающихся по самостоятельной работе

Самостоятельная работа обучающихся является составной частью учебной работы и имеет целью закрепление и углубление полученных знаний и навыков, поиск и приобретение новых знаний, в том числе с использованием автоматизированных обучающих курсов (систем), а также выполнение учебных заданий, подготовку к предстоящим занятиям, зачетам и экзаменам.

Самостоятельная работа во внеаудиторное время включает в себя:

- повторение лекционного материала;
 - подготовку к лабораторным работам (практическим занятиям);
 - изучение учебной и научной литературы;
 - решение задач, выданных на практических занятиях;
 - подготовку к тестированию;
 - выделение наиболее сложных и проблемных вопросов по изучаемой теме, получение разъяснений и рекомендаций по данным вопросам с преподавателями кафедры на их еженедельных консультациях.
- проведение самоконтроля путем ответов на вопросы текущего контроля знаний, решения представленных в учебно-методических материалах кафедры задач, тестов по отдельным вопросам изучаемой дисциплины.

Перед выполнением внеаудиторной самостоятельной работы преподаватель проводит инструктаж (консультацию) с определением цели задания, его содержания, сроков выполнения, основных требований к результатам работы, критериев оценки, форм контроля и перечня источников и литературы.

Обычно постановку задачи обучаемым на проведение самостоятельной работы преподаватель осуществляет на одном из занятий, предшествующем данному.

Методику самостоятельной работы все обучаемые выбирают индивидуально.