

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Ганеев Винер Валиахметович
Должность: Директор
Дата подписания: 04.06.2024 09:34:00
Уникальный программный ключ:
fceab25d7092f3bff743e8ad3f8d57fddc1f5e66

ФГБОУ ВО «УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»
БИРСКИЙ ФИЛИАЛ УУНиТ
ФАКУЛЬТЕТ ФИЗИКИ И МАТЕМАТИКИ

Утверждено:

на заседании кафедры информатики и
экономики
протокол № 4 от 24.11.2023 г.
Зав. кафедрой подписано ЭЦП /Мухаметшина Г.С.

Согласовано:

Председатель УМК
факультета физики и математики
подписано ЭЦП /Бигаева Л.А.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)
для очной формы обучения**

Специальные средства контроля и мониторинга информационных систем

Часть, формируемая участниками образовательных отношений

программа магистратуры

Направление подготовки (специальность)
09.04.03 *Прикладная информатика*

Направленность (профиль) подготовки
Информационные системы

Квалификация
Магистр

Разработчик (составитель) Старший преподаватель (должность, ученая степень, ученое звание)	<u>подписано ЭЦП /Гилев А.Ю.</u> (подпись, Фамилия И.О.)
--	---

Для приема: 2024-2025 г.

Бирск 2023 г.

Составитель / составители: Гилев А.Ю.

Рабочая программа дисциплины утверждена на заседании кафедры информатики и экономики протокол № ____ от « ____ » _____ 20__ г.

Дополнения и изменения, внесенные в рабочую программу дисциплины, утверждены на заседании кафедры _____, протокол № ____ от « ____ » _____ 20__ г.

Заведующий кафедрой _____ / _____ Ф.И.О/

Дополнения и изменения, внесенные в рабочую программу дисциплины, утверждены на заседании кафедры _____, протокол № ____ от « ____ » _____ 20__ г.

Заведующий кафедрой _____ / _____ Ф.И.О/

Дополнения и изменения, внесенные в рабочую программу дисциплины, утверждены на заседании кафедры _____, протокол № ____ от « ____ » _____ 20__ г.

Заведующий кафедрой _____ / _____ Ф.И.О/

Дополнения и изменения, внесенные в рабочую программу дисциплины, утверждены на заседании кафедры _____, протокол № ____ от « ____ » _____ 20__ г.

Заведующий кафедрой _____ / _____ Ф.И.О/

Список документов и материалов

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с установленными в образовательной программе индикаторами достижения компетенций.....	4
2. Цель и место дисциплины в структуре образовательной программы.....	6
3. Содержание рабочей программы (объем дисциплины, типы и виды учебных занятий, учебно-методическое обеспечение самостоятельной работы обучающихся).....	6
4. Фонд оценочных средств по дисциплине	12
4.1. Перечень компетенций и индикаторов достижения компетенций с указанием соотнесенных с ними запланированных результатов обучения по дисциплине. Описание критериев и шкал оценивания результатов обучения по дисциплине	12
4.2. Типовые контрольные задания или иные материалы, необходимые для оценивания результатов обучения по дисциплине, соотнесенных с установленными в образовательной программе индикаторами достижения компетенций. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине	14
4.3. Рейтинг-план дисциплины	21
5. Учебно-методическое и информационное обеспечение дисциплины	21
5.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины	21
5.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» и программного обеспечения, необходимых для освоения дисциплины	22
6. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине.....	22

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с установленными в образовательной программе индикаторами достижения компетенций

По итогам освоения дисциплины обучающийся должен достичь следующих результатов обучения:

Категория (группа) компетенций (при наличии ОПК)	Формируемая компетенция (с указанием кода)	Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине
	Способен оценивать качество и надежность прикладных информационных систем с учетом информационной безопасности на стадиях жизненного цикла. (ПК-1);	ПК-1.1. Оценивает качество прикладных информационных систем с учетом информационной безопасности	Знает способы оценки качества ИС; умеет применять способы оценки качества ИС.
		ПК-1.2. Оценивает надежность прикладных информационных систем с учетом информационной безопасности	Знает методы оценки надёжности ИС; владеет методами оценка надёжности ИС с учётом ИБ
		ПК-1.3. Разрабатывает систему контроля качества и надежности прикладных информационных систем с учетом информационной безопасности на стадиях жизненного цикла	Владеет способами разработки систем контроля качества и надёжности прикладных информационных систем.
	Способен управлять работами по сопровождению и проектами создания (модификации) информационных систем, автоматизирующих задачи организационного управления (ПК-2);	ПК-2.1. Планирует работы по проектированию, созданию и модификации информационных систем, автоматизирующих задачи организационного	Владеет навыками планирования процессов проектирования и разработки ИС
		ПК-2.2. Организует работы по проектированию, созданию и модификации информационных систем, автоматизирующих задачи организационного управления.	Умеет организовывать процесс разработки и выпуска ИС
		ПК-2.3. Осуществляет контроль и мониторинг работ по созданию, модификации и сопровождению информаци-	Умеет выполнять контроль и мониторинг процесса разработки и выпуска ИС

	онных систем, автоматизирующих задачи организационного управления.	
--	--	--

2. Цель и место дисциплины в структуре образовательной программы

Дисциплина «Специальные средства контроля и мониторинга информационных систем» относится к части, формируемой участниками образовательных отношений.

Дисциплина изучается на 2 курсе в 3,4 семестре.

Цель изучения дисциплины: формирование у магистрантов знаний, умений и навыков, необходимых для использования DevOps, мониторинга приложений микросервисной архитектуры.

3. Содержание рабочей программы (объем дисциплины, типы и виды учебных занятий, учебно-методическое обеспечение самостоятельной работы обучающихся)

ФГБОУ ВО «УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ»
БИРСКИЙ ФИЛИАЛ УУНиТ
ФАКУЛЬТЕТ ФИЗИКИ И МАТЕМАТИКИ

СОДЕРЖАНИЕ РАБОЧЕЙ ПРОГРАММЫ

дисциплины «Специальные средства контроля и мониторинга информационных систем» на _

3,4 семестр

очная

форма обучения

Вид работы	Объем дисциплины
Общая трудоемкость дисциплины (ЗЕТ / часов)	8/288
Учебных часов на контактную работу с преподавателем:	81.7
лекций	32
практических/ семинарских	0
лабораторных	48
контроль самостоятельной работы (КСР)	0
других (групповая, индивидуальная консультация и иные виды учебной деятельности, предусматривающие работу обучающихся с преподавателем) ФКР	1.7
Учебных часов на самостоятельную работу обучающихся (СРС)	171.5
Учебных часов на подготовку к экзамену (Контроль)	34.8

Форма контроля:

Экзамен 4 семестр

№ п/п	Тема и содержание	Форма изучения материалов: лекции, практические занятия, семинарские занятия, лабораторные работы, самостоятельная работа и трудоемкость (в часах)					Основная и дополнительная литература, рекомендуемая студентам (номера из списка)	Задания по самостоятельной работе студентов	Форма текущего контроля успеваемости (коллоквиумы, контрольные работы, компьютерные тесты и т.п.)
		Лек	Лаб	Эк	КоР	СР С			
2 курс / 3 семестр									
1	Микросервисы								
1.1	Введение в микросервисы Определение. Модульная архитектура. Преимущества и недостатки. Предпосылки и условия перехода к микросервисной архитектуре. Кривая обучения в организации. Составляющие затрат.	2				5	Осн. лит-ра № 1	Лабораторная работа	Лабораторная работа
1.2	Межпроцессные взаимодействия Типы взаимодействий. Подготовка к разработке веб-служб. Сопровождение микросервисов. Обнаружение службы.	2	2			5	Осн. лит-ра № 1	Лабораторная работа	Лабораторная работа
1.3	Миграция и реализация микросервисов Необходимые условия. Создание нового приложения на основе микросервисов. Переход от монолитной архитектуры к архитектуре микросервисов.	2	2			5	Осн. лит-ра № 1	Лабораторная работа	Лабораторная работа

2	Технология контейнеров								
2.1	Контейнеры Docker Виртуальные машины. Контейнеры. Архитектура и компоненты Docker. Установка Docker.	2	4			5	Осн. лит-ра № 1 Доп. лит-ра №№ 1,2	Лабораторная работа	Лабораторная работа
2.2	Интерфейс Docker Основные команды. Dockerfile. Компоновщик Docker Compose.	2	6			5	Осн. лит-ра № 1 Доп. лит-ра №№ 1,2	Лабораторная работа	Лабораторная работа
2.3	Поддержка сети в контейнерах. Организация контейнеров Прямое соединение. Варианты подключения по умолчанию. Kubernetes. Apache mesos и Marathon. Docker Swarm. Обнаружение служб. Реестр служб.	2	4			5	Осн. лит-ра № 1 Доп. лит-ра №№ 1,2	Лабораторная работа	Лабораторная работа
2.4	Управление контейнерами Мониторинг. Журналирование. Сбор параметров. Инструменты мониторинга кластеров.	2	6			3.5	Осн. лит-ра № 1 Доп. лит-ра №№ 1,2	Лабораторная работа	Лабораторная работа
3	Контрольная работа				1	0.5			
Итого по 2 курсу 3 семестру		14	24		1	34			
2 курс / 4 семестр									
1	DevOps								

1.1	<p>Определение DevOps</p> <p>Определение. Цели. Истоки возникновения. Бережливое производство. Agile.</p>	2	2			2	Осн. лит-ра № 2 Доп. лит-ра № 3	Лабораторная работа	Лабораторная работа
1.2	<p>Принципы DevOps</p> <p>Поток создания ценности. Конвейер развёртывания. Система контроля версий. Автоматизированное управление конфигурациями.</p>	2	2			2	Осн. лит-ра № 2 Доп. лит-ра № 3	Лабораторная работа	Лабораторная работа
1.3	<p>Основные практики</p> <p>Ключевые отличия от традиционных практик. Визуализация работы. Ограничение числа задач в работе. Уменьшение задач. Выполнение операционных требований. Раннее выявление и устранение дефектов. Приоритизация задач. Область применения и ограничения DevOps. Эволюционирующая архитектура.</p>	4	8			42	Осн. лит-ра № 2 Доп. лит-ра № 3	Лабораторная работа	Лабораторная работа
2	Сервисные сетки								
2.1	<p>Введение в сервисные сетки</p> <p>Определение. Экосистема. Преимущества и недостатки сервисных сеток. Сервисная сетка Istio. Облачный подход к равномерной наблюдаемости</p>	2	2			2	Осн. лит-ра № 3	Лабораторная работа	Лабораторная работа
2.2	Основы Istio	4	4			42	Осн. лит-ра № 3	Лабораторная работа	Лабораторная работа

	Архитектура. Уровни. Расширяемость. Масштабируемость и управляемость. Развёртывание.							
2.3	Расширенные возможности Istio Прокси для сервисов. Безопасность. Pilot. Управление трафиком. Mixer и политика в сетке. Телеметрия. Отладка.	4	6			48	Осн. лит-ра № 3	Лабораторная работа Лабораторная работа
3	Экзамен			1		36		
Итого по 2 курсу 4 семестру		18	24	1		174		
Итого по дисциплине		32	48	1	1	208		

4. Фонд оценочных средств по дисциплине

4.1. Перечень компетенций и индикаторов достижения компетенций с указанием соотнесенных с ними запланированных результатов обучения по дисциплине. Описание критериев и шкал оценивания результатов обучения по дисциплине.

Код и формулировка компетенции: Способен оценивать качество и надежность прикладных информационных систем с учетом информационной безопасности на стадиях жизненного цикла. (ПК-1);

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Критерии оценивания результатов обучения (Экзамен)			
		2 (Неудовлетворительно)	3 (Удовлетворительно)	4 (Хорошо)	5 (Отлично)
ПК-1.1. Оценивает качество прикладных информационных систем с учетом информационной безопасности	Знает способы оценки качества ИС; умеет применять способы оценки качества ИС.	Знания не сформированы	Знания недостаточно сформированы, несистемны	Знания сформированы, но имеют отдельные пробелы и неточности	Знания полностью сформированы
ПК-1.2. Оценивает надежность прикладных информационных систем с учетом информационной безопасности	Знает методы оценки надежности ИС; владеет методами оценка надежности ИС с учётом ИБ	Умения не сформированы	Умения не полностью сформированы	Умения в основном сформированы	Умения полностью сформированы
ПК-1.3. Разрабатывает систему контроля качества и надежности прикладных информационных систем с учетом информационной безопасности на стадиях жизненного цикла	Владеет способами разработки систем контроля качества и надежности прикладных информационных систем.	Владение навыками не сформировано	Владение навыками неуверенное	Владение навыками в основном сформировано	Владение навыками уверенное

Код и формулировка компетенции: Способен управлять работами по сопровождению и проектами создания (модификации) информационных систем, автоматизирующих задачи организационного управления (ПК-2);

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Критерии оценивания результатов обучения (Экзамен)			
		2 (Неудовлетворительно)	3 (Удовлетворительно)	4 (Хорошо)	5 (Отлично)
ПК-2.1. Планирует работы по проектированию, созданию и модификации информационных систем, автоматизирующих задачи организационного	Владеет навыками планирования процессов проектирования и разработки ИС	Знания не сформированы	Знания недостаточно сформированы, несистемны	Знания сформированы, но имеют отдельные пробелы и неточности	Знания полностью сформированы
ПК-2.2. Организует работы по проектированию, созданию и модификации информационных систем, автоматизирующих задачи организационного управления.	Умеет организовывать процесс разработки и выпуска ИС	Умения не сформированы	Умения не полностью сформированы	Умения в основном сформированы	Умения полностью сформированы
ПК-2.3. Осуществляет контроль и мониторинг работ по созданию, модификации и сопровождению информационных систем, автоматизирующих задачи организационного управления.	Умеет выполнять контроль и мониторинг процесса разработки и выпуска ИС	Владение навыками не сформировано	Владение навыками неуверенное	Владение навыками в основном сформировано	Владение навыками уверенное

Критериями оценивания являются баллы, которые выставляются за виды деятельности (оценочные средства) по итогам изучения модулей (разделов дисциплины), перечисленных в рейтинг-плане дисциплины. Баллы, выставляемые за конкретные виды деятельности представлены ниже.

4.2. Типовые контрольные задания или иные материалы, необходимые для оценивания результатов обучения по дисциплине, соотнесенных с установленными в образовательной про-

грамме индикаторами достижения компетенций. Методические материалы, определяющие процедуры оценивания результатов обучения по дисциплине.

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные средства
ПК-1.1. Оценивает качество прикладных информационных систем с учетом информационной безопасности	Знает способы оценки качества ИС; умеет применять способы оценки качества ИС.	Контрольная работа, Лабораторная работа
ПК-1.2. Оценивает надежность прикладных информационных систем с учетом информационной безопасности	Знает методы оценки надёжности ИС; владеет методами оценка надёжности ИС с учётом ИБ	Контрольная работа, Лабораторная работа
ПК-1.3. Разрабатывает систему контроля качества и надёжности прикладных информационных систем с учетом информационной безопасности на стадиях жизненного цикла	Владеет способами разработки систем контроля качества и надёжности прикладных информационных систем.	Лабораторная работа, Контрольная работа
ПК-2.1. Планирует работы по проектированию, созданию и модификации информационных систем, автоматизирующих задачи организационного	Владеет навыками планирования процессов проектирования и разработки ИС	Лабораторная работа, Контрольная работа
ПК-2.2. Организует работы по проектированию, созданию и модификации информационных систем, автоматизирующих задачи организационного управления.	Умеет организовывать процесс разработки и выпуска ИС	Лабораторная работа, Контрольная работа
ПК-2.3. Осуществляет контроль и мониторинг работ по созданию, модификации и сопровождению информационных систем, автоматизирующих задачи организационного управления.	Умеет выполнять контроль и мониторинг процесса разработки и выпуска ИС	Контрольная работа, Лабораторная работа

Критериями оценивания при модульно-рейтинговой системе являются баллы, которые выставляются преподавателем за виды деятельности (оценочные средства) по итогам изучения модулей (разделов дисциплины), перечисленных в рейтинг-плане дисциплины

для экзамена: текущий контроль – максимум 40 баллов; рубежный контроль – максимум 30 баллов, поощрительные баллы – максимум 10;

Шкалы оценивания:

для экзамена:

от 45 до 59 баллов – «удовлетворительно»;

от 60 до 79 баллов – «хорошо»;

от 80 баллов – «отлично».

Контрольная работа

Создайте директорию Docker на рабочем столе.

Задача 1

Перейдите в директорию Docker и создайте в ней папку nginx и перейдите в нее. Создайте в ней две папки www и conf.d.

Создайте образ с nginx используя docker file на основе образа контейнера ubuntu 18.04. Пробросьте 80 порт из контейнера в локальную сеть на порт 80.

Примонтируйте локальную директорию .../Docker/nginx/conf.d в директорию контейнера /etc/nginx/conf.d.

Примонтируйте директорию .../Docker/nginx/www в директорию контейнера /var/www.

Создайте простой конфиг виртуального хоста для сайта в папке .../Docker/nginx/conf.d укажите server_name как nginx.local.

```
server{ listen 80; root /var/www/mysite; index index.html index.htm index.nginx-debian.html; server_name nginx.local;}
```

Создайте на локальном ПК в папке www директорию mysite. Скачайте какойнибудь bootstrap сайт и переместите в созданную директорию.

Измените файл **hosts** на локальном ПК так чтобы сайт открывался по доменному имени nginx.local. Учитывайте что для проверки работает ли у вас сайт без доменного имени вам понадобится либо сменить порт работы сайта и соответственно другой порт пробрасывать в локальную сеть либо же удалить стандартный сайт nginx.

Для создания и запуска контейнера на основе docker file вы должны использовать Docker-compose.

Задача 2

Перейдите в директорию Docker и создайте в ней папку apache и перейдите в нее. Создайте в ней две папки www и conf.d.

В папке www создайте файл index.html с следующим содержимым:

```
<h1>Привет от (ваше имя) из группы (ваша группа)/h1</h1>
```

Используя docker file создайте образ с apache в котором примонтируйте локальные директории так чтобы apache мог прочитать конфиги в conf.d на локальном пк и открыть сайт лежащий в www на локальном пк.

Создайте простой конфиг виртуального хоста apache.

По доменному имени apache.local у вас должен открываться сайт.

Для создания и запуска контейнера на основе docker file вы должны использовать Docker-compose.

Задача 3

Перейдите в директорию Docker и создайте в ней папку Lemp и перейдите в нее.

Создайте 2 docker file для двух контейнеров соответственно с php + nginx и mysql. Свяжите все так чтобы стек работал.

Для запуска стека вы должны использовать один Docker-compose файл

Задача 4

Напишите простой микросервис, калькулятор принимающий (число:действие:число). Пусть сервис принимает строки в json и выдает в ответ результат в json.

Напишите второй микросервис который будет получать какой-то текст и отправлять его вам в телеграм через бота.

Напишите веб приложений (сайт), который будет принимать выражение из 2 аргументов и показывать результат. Под капотом веб приложение должно считывать запрос пользователя с сайта и отправлять запрос на 1 микросервис, получая ответ веб приложение должно показать ответ пользователю на сайте и отправить полученное число на второй микросервис.

Создайте 3 docker file в которых опишите среды в которых будут запускаться ваши микропрограммы.

Пробросьте все необходимые порты для работы ваших сетевых приложений.

Методические материалы, определяющие процедуру оценивания контрольной работы

Описание методики оценивания: при оценке выполнения студентом контрольной работы максимальное внимание следует уделять следующим аспектам: насколько полно в теоретическом вопросе раскрыто содержание материала, четко и правильно даны определения, раскрыто содержание понятий; верно использованы научные термины; демонстрируются высокий уровень умения оперировать научными категориями, анализировать информацию, владение навыками практической деятельности; кейс-задание решено на высоком уровне, содержит аргументацию и пояснения.

Критерии оценки (в баллах):

- **9-10** баллов выставляется студенту, если в теоретическом вопросе полно раскрыто содержание материала; четко и правильно даны определения, раскрыто содержание понятий; верно использованы научные термины; демонстрируются высокий уровень умения оперировать научными категориями, анализировать информацию, владение навыками практической деятельности; кейс-задание решено на высоком уровне, содержит пояснения; тестовые задания решены свыше, чем на 80%; уровень знаний, умений, владений – высокий;
- **7-8** баллов выставляется студенту, если в теоретическом вопросе раскрыто основное содержание материала; в основном правильно даны определения понятий и использованы научные термины; ответ самостоятельный; определения понятий неполные, допущены незначительные нарушения в последовательности изложения; небольшие недостатки при использовании научных терминов; кейс-задание решено верно, но решение не доведено до завершающего этапа; тесты решены на 60-80%. Уровень знаний, умений, владений – средний;
- **5-6** баллов выставляется студенту, если в теоретическом вопросе усвоено основное, но непоследовательно; определения понятий недостаточно четкие; не использованы в качестве доказательства выводы и обобщения из наблюдений, практических занятий; уровень умения оперировать научными категориями, анализировать информацию, владения навыками практической деятельности невысокий, наблюдаются пробелы и неточности; в решение кейс-задания верно выполнены некоторые этапы; тесты решены на 40-60%; уровень знаний, умений, владений – удовлетворительный;
- **менее 5** баллов выставляется студенту, если в теоретическом вопросе не изложено основное содержание учебного материала, изложение фрагментарное, не последовательное; определения понятий не четкие; не использованы в качестве доказательства выводы и обобщения из наблюдений, уровень умения оперировать научными категориями, анализировать информацию, владения навыками практической деятельности очень низкий; тесты решены менее, чем на 40 %; уровень знаний, умений, владений – недостаточный.

Лабораторная работа

Лаб. раб. 1. Основы docker

Введение

Целью данной лабораторной работы является знакомство с программой docker.

Установка docker

Прочитайте про инструментарий для установки программ [Advanced Packaging Tool](#) и консольный интерфейс к этому инструментарию [Apt-Get](#).

Далее, в связи с изменением инструкции по установке docker на сайте docker, будет рассмотрен способ установки, отличный от предлагаемого ранее на этой странице. Если Вы уже выполнили

эту лабораторную работу с помощью старого способа, то при оформлении отчета и защиты работы можете его использовать, его описание – под спойлером.

Первоначальный способ установки. В терминале виртуальной машины выполните команду, которая обновит информацию о доступных для установки программах:

```
sudo apt-get update
```

Установите дополнительные пакеты, которые потребуются для установки docker:

```
sudo apt-get install \ apt-transport-https \ ca-certificates \ curl \ gnupg-agent \ software-properties-common
```

Установите цифровой ключ для проверки подписи устанавливаемого программного обеспечения (docker):

```
curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
```

Установите репозиторий docker:

```
sudo add-apt-repository \ "deb [arch=amd64] https://download.docker.com/linux/ubuntu \ $(lsb_release -cs) \ stable"
```

Установите движок docker:

```
sudo apt-get updatesudo apt-get install docker-ce docker-ce-cli containerd.io
```

Перейдите на страницу <https://download.docker.com/linux/ubuntu/dists/>, выберите нужную версию Ubuntu, перейдите в папку /pool/stable, выберите архитектуру amd64, armhf или arm64 и загрузите последние версии файлов containerd.io_□□□□□□□□.deb, docker-ce-cli_□□□□□□□□.deb и docker-ce_□□□□□□□□.deb. Для рекомендуемой ранее виртуальной машины эти файлы находятся по адресу <https://download.docker.com/linux/ubuntu/dists/focal/pool/stable/amd64/>.

Установите каждый из скаченных файлов с помощью команды:

```
sudo dpkg -i □□□□□□□□.deb
```

Проверьте корректность установки:

```
sudo docker run hello-world
```

Работа с образами

Посмотрите имеющиеся в репозитории docker образы с помощью команды

```
sudo docker images
```

Попробуйте удалить имеющийся образ с помощью команды

```
sudo docker image rm □□□□□□_□□
```

Посмотрите контейнеры с помощью команды (ключ -a – показывать остановленные контейнеры)

```
sudo docker ps -a
```

Удалите контейнер с помощью команды

```
sudo docker container rm □□□□□□□□□□_□□ или □□□□□□
```

Удалите образ.

Скачайте образ с операционной системой alpine:

```
sudo docker pull alpine
```

Запустите образ alpine. Посмотрите список контейнеров.

Выполните команду

```
sudo docker run -it alpine
```

Запустите еще один терминал на виртуальной машине и посмотрите в нем список контейнеров.

Удалите контейнер с запущенным образом alpine. Посмотрите, что произошло в первом терминале.

Лаб. раб. 2. Настройка ресурсов контейнеров docker

Введение

Целью данной лабораторной работы является изучение взаимодействия контейнеров docker с операционной системой.

Создание общей папки для контейнера

Скачайте образ docker, содержащий средства разработки под linux:

```
sudo docker pull gcc
```

Запустите этот контейнер указав следующие ключи:

```
-it -v /home/osboxes/server:/home/osboxes
```

Ключ `-v` создает общую папку, доступную на виртуальной машине по пути `/home/osboxes/server`, а в контейнере по пути `/home/osboxes`.

На физическом компьютере или виртуальной машине создайте файл `hello.c` со следующим содержанием:

```
#include "stdio.h" main () { #if defined(_WIN32) printf("hello, windows\n"); #elif defined(__linux__) printf("hello, linux\n"); #elif defined(__APPLE__) printf("hello, Apple\n"); #elif defined(BSD) printf("hello, BSD\n"); #endif }
```

Скомпилируйте и запустите эту программу на физическом компьютере и в контейнере. Для запуска в контейнере нужно перейти в его терминале в общую папку, запустить компилятор `gcc`, и запустить результат компиляции командой `./a.out`.

Скопируйте любую лабораторную работу из 1-го семестра по программированию в расшаренную папку, скомпилируйте ее в контейнере и запустите.

Проброс портов

Скачайте образ `gitbucket` (сервер системы контроля версий):

```
sudo docker pull gitbucket/gitbucket
```

При запуске контейнера ключ `-it` указывать не нужно, нужно указать ключи `-v` и `-p`.

Создайте на физической машине папку для хранения данных `gitbucket` и сделайте ее с помощью ключа `-v` общей с папкой контейнера `/gitbucket` (через общую папку на виртуальной машине).

Ключ `-p` нужен для проброса портов. Он имеет формат:

```
-p [host_ip]_[host_port]:[container_ip]_[container_port]
```

Сервер в контейнере запускается на порту `8080`, этот порт следует указать в

```
[host_ip]_[host_port]_[container_ip]_[container_port]. [host_ip]_[host_port] – это порт, на котором будет доступен сервер на виртуальной машине (его нужно пробросить на физическую машину).
```

Дождитесь загрузки сервера (в терминале должен появиться текст `Server:main: Started`). В браузере компьютера зайдите на сервер `gitbucket`, аутентифицируйтесь под логином `root` с паролем `root`. Создайте репозиторий системы контроля версий с настройками по-умолчанию.

Перегрузите виртуальную машину, перезапустите контейнер и убедитесь, что учетная запись и репозиторий на сервере сохранились.

Взаимодействие контейнеров

По-умолчанию, контейнеры `docker` подключаются к виртуальной компьютерной сети через которую и осуществляется их взаимодействие. Каждый контейнер в этой сети имеет свой адрес.

В этом пункте мы будем с контейнера со средствами разработки выкладывать результаты на сервер контроля версий. Для этого нужно узнать его адрес (порт сервера уже известен – `8080`). Для этого:

- с помощью команды `sudo docker ps` определите имя контейнера с сервером.
- с помощью команды `sudo docker network inspect bridge` посмотрите конфигурацию виртуальной сети. В этой конфигурации в разделе `Containers` найдите контейнер с нужным именем. В параметрах этого контейнера будет примерно такая строка `"IPv4Address": "172.17.0.3/16"`, где `172.17.0.3` – интересующий нас адрес контейнера сервера.

Далее создадим локальную копию репозитория и выложим ее на сервер. В терминале контейнера `gcc` выполните следующие действия:

- настройте пользователя системы контроля версий:

```
git config --global user.email "[host_ip]@[container_ip].[host_ip]" git config --global user.name "[host_ip] [host_ip]"
```
- создайте и перейдите в каталог для локального репозитория:

```
mkdir repocd repo
```
- создайте репозиторий и привяжите его к репозиторию на сервере:

```
touch README.md git init git add README.md git commit -m "first commit" git remote add origin http://[host_ip]_[host_port]:[container_ip]_[container_port]/git/root/[host_ip]-[host_ip].git git push -u origin master
```

Вместо □□□□-□□□□ укажите имя созданного ранее репозитория. Выполнение последней команды потребует ввода имени пользователя и пароля на сервере системы контроля версий. Используйте тот же логин, что и ранее (root:root).

Перейдите в браузере на страницу репозитория и убедитесь, что в нем появился файл README.md.

Методические материалы, определяющие процедуру оценивания выполнения лабораторных работ

Процедура сдачи лабораторной работы состоит из

1. Теоретической части, которая состоит из 2-х вопросов (низкой или средней сложности), ответы на которые оцениваются на 0-2 балла. Переход к сдаче практической части возможен только если за теоретическую часть получено не менее 2 баллов.

2. Практической части, которая содержит несколько задач (обычно 3-5), за каждую из которых можно получить от 2 до 5 баллов.

Суммарно по всем частям можно получить до 20 баллов. Эта оценка, наравне с оценками за другие лабораторные работы, используется при расчёте рейтинга по дисциплине.

Параллельно с этим, для удобства её можно перевести в 4-х балльную оценку, аналогично тому, как это делается для всей дисциплины (неудовлетворительно: < 9 баллов, удовлетворительно: 9 - 13, хорошо: 14 - 15, отлично 16 - 20).

Экзаменационные билеты

Экзамен (зачет) является оценочным средством для всех этапов освоения компетенций. Структура экзаменационного билета: в билете указывается кафедра в рамках нагрузки которой реализуется данная дисциплина, форма обучения, направление и профиль подготовки, дата утверждения; билет может включать в себя теоретический(ие) вопрос(ы) и практическое задание (кейс-задание).

Примерные вопросы к экзамену, 2 курс / 4 семестр

1. Микросервисы. Определение. Преимущества и недостатки.
2. Межпроцессные взаимодействия. Типы. Сопровождение микросервисов. Обнаружение служб.
3. Контейнеры Docker. Виртуальные машины. Контейнеры. Архитектура и компоненты.
4. Docker. Команды search, pull, images, rmi, run, ps, logs, restart.
5. Docker. Команды attach, rm, inspect, exec, rename, cp, pause/unpause, create, commit, diff.
6. Dockerfile. Компоновщик Docker Compose.
7. Организация контейнеров: Kubernetes.
8. Организация контейнеров: Apache Mesos и Marathon.
9. Организация контейнеров: Docker Swarm.
10. Управление контейнерами: Мониторинг. Журналирование. Сбор параметров.
11. Инструменты мониторинга контейнеров.
12. DevOps: Определение. Цели. История возникновения.
13. Принципы DevOps: Поток создания ценности. Конвейер развёртывания.
14. Принципы DevOps: Использование системы контроля версий. Автоматизированное управление конфигурациями. Определение завершения.
15. Основные практики DevOps: Отличия от традиционных практик. Визуализация работы.
16. Основные практики DevOps: Ограничения числа задач в работе. Уменьшение размера задач.
17. Основные практики DevOps: Приоритизация задач. Устранение узких мест.
18. Область применения и ограничения DevOps. Эволюционирующая архитектура.
19. Сервисные сетки. Определение. Экосистема. Преимущества сервисных сеток.
20. Istio: Архитектура сервисной сетки. Уровни. Расширяемость. Масштабируемость и производительность.
21. Istio: Прокси для сервисов.
22. Istio: Безопасность и идентичность.

23. Istio: Pilot. Управление трафиком.

24. Istio: Телеметрия и отладка.

Образец экзаменационного билета

МИНОБРНАУКИ РФ ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «УФИМСКИЙ УНИВЕРСИТЕТ НАУКИ И ТЕХНОЛОГИЙ» БИРСКИЙ ФИЛИАЛ УУНиТ Кафедра информатики и экономики	
Дисциплина: Специальные средства контроля и мониторинга информационных систем очная форма обучения 2 курс 4 семестр	Курсовые экзамены 20__-20__ г. Направление 09.04.03 Прикладная информатика Профиль: Информационные системы
Экзаменационный билет № 1 1. Межпроцессные взаимодействия. Типы. Сопровождение микросервисов. Обнаружение служб. 2. Принципы DevOps: Использование системы контроля версий. Автоматизированное управление конфигурациями. Определение завершения.	
Дата утверждения: __.__._____	Заведующий кафедрой _____

Методические материалы, определяющие процедуру оценивания ответа на экзамене

Критериями оценивания являются баллы, которые выставляются за виды деятельности (оценочные средства) по итогам изучения модулей (разделов дисциплины), перечисленных в рейтинг-плане дисциплины: текущий контроль – максимум 40 баллов; рубежный контроль – максимум 30 баллов, поощрительные баллы – максимум 10.

При оценке ответа на экзамене максимальное внимание должно уделяться тому, насколько полно раскрыто содержание материала, четко и правильно даны определения, раскрыто содержание понятий, верно ли использованы научные термины, насколько ответ самостоятельный, использованы ли ранее приобретенные знания, раскрыты ли раскрыты причинно-следственные связи, насколько высокий уровень умения оперирования научными категориями, анализа информации, владения навыками практической деятельности.

Критерии оценки (в баллах):

- **25-30 баллов** выставляется студенту, если студент дал полные, развернутые ответы на все теоретические вопросы билета, продемонстрировал знание функциональных возможностей, терминологии, основных элементов, умение применять теоретические знания при выполнении практических заданий. Студент без затруднений ответил на все дополнительные вопросы. Практическая часть работы выполнена полностью без неточностей и ошибок;

- **17-24 баллов** выставляется студенту, если студент раскрыл в основном теоретические вопросы, однако допущены неточности в определении основных понятий. При ответе на дополнительные вопросы допущены небольшие неточности. При выполнении практической части работы допущены несущественные ошибки;

- **10-16 баллов** выставляется студенту, если при ответе на теоретические вопросы студентом допущено несколько существенных ошибок в толковании основных понятий. Логика и полнота ответа страдают заметными изъянами. Заметны пробелы в знании основных методов. Теоретические вопросы в целом изложены достаточно, но с пропусками материала. Имеются принципиальные

ошибки в логике построения ответа на вопрос. Студент не решил задачу или при решении допущены грубые ошибки;

- **1-10 баллов** выставляется студенту, если ответ на теоретические вопросы свидетельствует о непонимании и крайне неполном знании основных понятий и методов. Обнаруживается отсутствие навыков применения теоретических знаний при выполнении практических заданий. Студент не смог ответить ни на один дополнительный вопрос.

Перевод оценки из 100-балльной в четырехбалльную производится следующим образом:

- отлично – от 80 до 110 баллов (включая 10 поощрительных баллов);

- хорошо – от 60 до 79 баллов;

- удовлетворительно – от 45 до 59 баллов;

- неудовлетворительно – менее 45 баллов.

4.3. Рейтинг-план дисциплины

Таблица перевода баллов текущего контроля в баллы рейтинга

	5	5	5	5	5	5	5	5	5	5
0	0	0	0	0	0	0	0	0	0	0
1	5	3	2	2	1	1	1	1	1	1
2		5	4	3	2	2	2	2	2	1
3			5	4	3	3	3	2	2	2
4				5	4	4	3	3	3	2
5					5	5	4	4	3	3
6						5	5	4	4	3
7							5	5	4	4
8								5	5	4
9									5	5
10										5

Рейтинг-план дисциплины представлен в Приложении 1.

5. Учебно-методическое и информационное обеспечение дисциплины

5.1. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины

Основная литература

1. Кочер, П. С. Микросервисы и контейнеры Docker : руководство / П. С. Кочер ; перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2019. — 240 с. — ISBN 978-5-97060-739-8. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/123710> (дата обращения: 09.01.2023). — Режим доступа: для авториз. пользователей.
2. Скрынник, О. В. DevOps для ИТ-менеджеров: концентрированное структурированное изложение передовых идей / О. В. Скрынник. — 2-е изд. — Москва : ДМК Пресс, 2019. — 126 с. — ISBN 978-5-97060-692-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/112933> (дата обращения: 09.01.2023). — Режим доступа: для авториз. пользователей.
3. Калькот, Л. Istio: приступаем к работе : руководство / Л. Калькот, З. Бутчер ; перевод с английского А. Л. Бриня. — Москва : ДМК Пресс, 2020. — 236 с. — ISBN 978-5-97060-863-0. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/179499> (дата обращения: 09.01.2023). — Режим доступа: для авториз. пользователей.

Дополнительная литература

1. Маркелов, А. А. Введение в технологию контейнеров и Kubernetes / А. А. Маркелов. — Москва : ДМК Пресс, 2019. — 194 с. — ISBN 978-5-97060-775-6. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/131702> (дата обращения: 09.01.2023). — Режим доступа: для авториз. пользователей.
2. Сейерс, Э. Х. Docker на практике / Э. Х. Сейерс, А. Милл ; перевод с английского Д. А. Беликов. — Москва : ДМК Пресс, 2020. — 516 с. — ISBN 978-5-97060-772-5. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/131719> (дата обращения: 09.01.2023). — Режим доступа: для авториз. пользователей.
3. Херинг, М. DevOps для современного предприятия : учебное пособие / М. Херинг ; перевод с английского М. А. Райтмана. — Москва : ДМК Пресс, 2020. — 232 с. — ISBN 978-5-97060-836-4. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/140580> (дата обращения: 09.01.2023). — Режим доступа: для авториз. пользователей.

5.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет» и программного обеспечения, необходимых для освоения дисциплины

1. Научная электронная библиотека eLIBRARY.RU [Электронный ресурс]. — Режим доступа: <https://elibrary.ru/>.
2. Электронная библиотечная система «Лань» [Электронный ресурс]. — Режим доступа: <https://e.lanbook.com/>.
3. Университетская библиотека онлайн biblioclub.ru [Электронный ресурс]. — Режим доступа: <http://biblioclub.ru/>.
4. Электронная библиотека УУНиТ [Электронный ресурс]. — Режим доступа: <https://elib.bashedu.ru/>.
5. Российская государственная библиотека [Электронный ресурс]. — Режим доступа: <https://www.rsl.ru/>.
6. Национальная электронная библиотека [Электронный ресурс]. — Режим доступа: <https://xn--90ax2c.xn--plai/viewers/>.
7. Национальная платформа открытого образования proed.ru [Электронный ресурс]. — Режим доступа: <http://npoed.ru/>.
8. Электронное образование Республики Башкортостан [Электронный ресурс]. — Режим доступа: <https://edu.bashkortostan.ru/>.
9. Информационно-правовой портал Гарант.ру [Электронный ресурс]. — Режим доступа: <http://www.garant.ru/>.

Программное обеспечение

1. Windows - Договор №0301100003620000022 от 29.06.2020, Договор № 2159- ПО/2021 от 15.06.2021, Договор №32110448500 от 30.07.2021
2. Браузер Google Chrome - Бесплатная лицензия https://www.google.com/intl/ru_ALL/chrome/privacy/eula_text.html
3. Office Professional Plus - Договор №0301100003620000022 от 29.06.2020, Договор № 2159- ПО/2021 от 15.06.2021, Договор №32110448500 от 30.07.2021
4. Файловый менеджер DoubleCommander - Бесплатная лицензия <https://sourceforge.net/projects/doublecmd/>

6. Материально-техническая база, необходимая для осуществления образовательного процесса по дисциплине

Наименование специализированных аудиторий, кабинетов, лабораторий	Вид занятий	Наименование оборудования, программного обеспечения
Аудитория 222(ФМ)	Для самостоятельной работы	Компьютеры в сборе, проектор, учебная мебель, экран для проекторов. Программное обеспечение <ol style="list-style-type: none"> 1. Office Professional Plus 2. Windows 3. Файловый менеджер DoubleCommander
Аудитория 301 Читальный зал (электронный каталог)(ФМ)	Для самостоятельной работы	Компьютеры в сборе, принтер, сканер, учебная мебель, учебно-методические материалы. Программное обеспечение <ol style="list-style-type: none"> 1. Браузер Google Chrome 2. Office Professional Plus 3. Windows
Аудитория 302(ФМ)	Лекционная, Семинарская	интерактивная доска, проектор, учебная мебель.
Аудитория 307(ФМ)	Лекционная	Нетбук, учебная мебель, экран Программное обеспечение <ol style="list-style-type: none"> 1. Office Professional Plus 2. Windows 3. Браузер Google Chrome
Аудитория 311(ФМ)	Для курсового проектирования, Для консультаций, Для контроля и аттестации, Для лабораторных занятий	Доска маркерная, компьютеры в сборе, мультимедийный проектор, экран настенный. Программное обеспечение <ol style="list-style-type: none"> 1. Браузер Google Chrome 2. Windows
Аудитория 313(ФМ)	Лекционная, Для консультаций, Для контроля и аттестации, Для лабораторных занятий	Интерактивная доска , компьютеры в комплекте, проектор, экран. Программное обеспечение <ol style="list-style-type: none"> 1. Браузер Google Chrome