

**Федеральное государственное автономное образовательное
учреждение высшего образования
«Московский физико-технический институт
(национальный исследовательский университет)»**

УТВЕРЖДЕНО

**Директор физтех-школы
прикладной математики и
информатики**

А.М. Райгородский

	Рабочая программа дисциплины (модуля)
по дисциплине:	Теория отказоустойчивых распределенных систем
по направлению:	Информатика и вычислительная техника
профиль подготовки:	Физтех-школа Прикладной Математики и Информатики кафедра алгоритмов и технологий программирования
курс:	3
квалификация:	бакалавр

Семестр, формы промежуточной аттестации: 5 (осенний) - Дифференцированный зачет

Аудиторных часов: 60 всего, в том числе:

лекции: 30 час.

семинары: 30 час.

лабораторные занятия: 0 час.

Самостоятельная работа: 75 час.

Всего часов: 135, всего зач. ед.: 3

Программу составил: Р.Г. Липовский, старший преподаватель

Программа обсуждена на заседании кафедры алгоритмов и технологий программирования 04.06.2020

Аннотация

Курс посвящен теории, лежащей в основе современных промышленных распределенных систем: файловых систем, очередей сообщений, key/value хранилищ, баз данных. Эти системы хранят десятки и сотни петабайт данных, обслуживают многие тысячи запросов в секунду и масштабируются до сотен и тысяч машин, переживая при этом отказы дисков и питания, дрейф часов, задержки и нарушения связности сети, а потому устроены невероятно сложно.

Но если посмотреть сквозь все инженерные детали и сотни тысяч строк кода, то окажется, что сложность, связанную с распределенностью, можно заключить в относительно простые модели и задачи: как узлам договориться о порядке доставки сообщений в асинхронной сети, как выбрать лидера среди равноправных машин, как добавить в систему еще один сервер или обнаружить сбойную машину. Именно от решения этих задач в конечном итоге будут зависеть важнейшие характеристики всей системы: границы ее отказоустойчивости, доступность при нестабильном поведении сети и модель согласованности данных.

В курсе мы рассмотрим эти задачи, исследуем ограничения, которые накладывает на них модель сети и сбоев, и потрогаем практические алгоритмы, которые применяются в известных промышленных распределенных системах.

1. Цели и задачи

Цель дисциплины

Научить студента видеть за распределенными системами ряд фундаментальных задач, которые определяют ключевые характеристики этих систем: отказоустойчивость, масштабируемость, доступность

Изучить различные модели сети и сбоев, исследовать ограничения, которые они накладывают на решения этих задач

Изучить ключевые алгоритмы, которые используются в промышленных распределенных системах

Научить студента ориентироваться в научной области, познакомиться с ключевыми академическими работами

Задачи дисциплины

Знает теоретические модели, ключевые задачи и результаты о невозможности (Atomic Broadcast, Consensus)

Знает алгоритмы, которые используются в промышленных распределенных системах (Multi-Paxos, RAFT, распределенные транзакции, PBFT, Bitcoin)

Знает подходы к верификации распределенных систем, владеет формальными методами верификации

Умеет программировать изученные алгоритмы с применением современных инструментов асинхронного программирования (файберы, фьючи и т.д.)

2. Перечень формируемых компетенций

Освоение дисциплины направлено на формирование следующих компетенций:

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности
	ОПК-4.4 Владеет навыками работы с компьютером и компьютерными сетями с целью получения, хранения и обработки научной (технической, технологической) информации
	ОПК-4.3 Умеет составлять аннотации, рефераты, библиографические перечни и обзоры информации в области своей профессиональной деятельности

ОПК-5 Способен участвовать в проведении фундаментальных и прикладных исследований и разработок, самостоятельно осваивать новые теоретические, в том числе, математические методы исследований и работать на современной экспериментальной научно-исследовательской, измерительно-аналитической и технологической аппаратуре)	ОПК-5.1 Способен решать поставленные задачи в области теоретических и экспериментальных исследований и разработок
	ОПК-5.2 Обладает способностью к освоению новых знаний на основе изучения литературы, научных статей и других источников
	ОПК-5.3 Способен к профессиональной эксплуатации современной экспериментальной научно-исследовательской (измерительно-аналитической и технологической) аппаратуры
ПК-1 Способен ставить, формализовывать и решать задачи, в том числе разрабатывать и исследовать математические модели изучаемых явлений и процессов, системно анализировать научные проблемы, получать новые научные результаты	ПК-1.1 Способен находить, анализировать и обобщать информацию об актуальных результатах исследований в рамках тематической области своей профессиональной деятельности
	ПК-1.2 Способен выдвигать гипотезы, строить математические модели для описания изучаемых явлений и процессов, оценить качество разработанной модели
	ПК-1.3 Способен применять теоретические и (или) экспериментальные методы исследований к конкретной научной задаче и интерпретировать полученные результаты
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации
	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива
	ПК-2.3 Способен проводить апробацию результатов научно-исследовательской работы посредством публикации научных статей и участия в конференциях

3. Перечень планируемых результатов обучения по дисциплине (модулю)

В результате освоения дисциплины обучающиеся должны

знать:

- Алгоритмы, решающие задачи вычислительной геометрии.
- Методики распараллеливания алгоритмов, способы синхронизации потоков, разделения доступа к данным и контроля исполнения подзадач.
- Оценки сложности стандартных алгоритмов.

уметь:

- Реализовывать алгоритмы, решающие задачи вычислительной геометрии.
- Реализовывать параллельные алгоритмы различной, выполнять синхронизацию потоков и доступа к данным.

владеть:

- Средствами стандартной библиотеки C++ для создания многопоточных приложений.
- Методами декомпозиции задач в области информационных технологий и построения единого решения с использованием изученных алгоритмов.

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

4.1. Разделы дисциплины (модуля) и трудоемкости по видам учебных занятий

		Трудоемкость по видам учебных занятий, включая самостоятельную работу, час.
--	--	---

№	Тема (раздел) дисциплины	Лекции	Семинары	Лаборат. работы	Самост. работа
1	Модель распределенной системы	4	4		9
2	Репликация и модели согласованности	4	4		6
3	Atomic Broadcast и State Machine Replication	2	4		6
4	Распределенный консенсус	2	2		6
5	Алгоритм Single Decree Paxos	2	2		6
6	Алгоритм Multi-Paxos	2	2		6
7	Paxos Made Live	2	2		6
8	Распределенные транзакции.	2	2		6
9	Формальные методы для верификации распределенных систем	2	2		6
10	Византийские отказы	2	2		6
11	Алгоритм Practical Byzantine Fault Tolerance (PBFT)	2	2		6
12	Bitcoin. Алгоритм HotStuff	4	2		6
Итого часов		30	30		75
Подготовка к экзамену		0 час.			
Общая трудоёмкость		135 час., 3 зач.ед.			

4.2. Содержание дисциплины (модуля), структурированное по темам (разделам)

Семестр: 5 (Осенний)

1. Модель распределенной системы

Модель распределенной системы: снаружи – внутри – узлы и модель передачи сообщений, снаружи – конкурентный атомарный объект. Моделирование узлов, сети, отказов, часов.

Время, виды часов (кварцевые, атомные), дрейф. Невозможность синхронизации часов без дрейфа в синхронной сети с неопределенностью при доставке сообщений. GPS и синхронизация часов, применение GPS и атомных часов в распределенных системах: TrueTime.

Семинар:

Время в промышленных системах: монотонные и wall time часы, стандартные временные оси, NTP и монотонность, високосные секунды и leap smearing.

Устройство датацентра: стойки и ToR коммутаторы, коммутационная файбрика, отказоустойчивость и задержки внутри ДЦ, охлаждение и питание. Магистральные кабели: задержки, поломки. Google private network и партишены.

Понятие failure domains

2. Репликация и модели согласованности

Мотивирующий пример: K/V хранилище, репликация, модели согласованности, линейаризуемость.

Задача репликации регистра, алгоритм ABD. Наивные кворумные операции и нарушение линейаризуемости. Случай одного писателя, двухфазное чтение, аналогия со свободой от блокировок в многопоточных алгоритмах, док-во линейаризуемости. Случай многих писателей, согласованный выбор временных меток, физическое время / двухфазная запись, док-во линейаризуемости. Выбор монотонных меток с помощью TrueTime. Переконфигурация набора реплик, рестарты реплик, возможность обобщения на более сложные операции.

Семинар:

Устройство отказоустойчивого локального хранилища для key/value хранилища / базы данных. Выбор API, выбор гарантий надежности: atomicity, durability. Выбор модели оценки сложности.

Физические устройства: HDD, SSD. HDD: время поворота блина и время seek-a, паттерны доступа и их стоимости. Характеристики современных HDD. SSD: API флэш памяти, страницы и блоки, изнашивание, FTL, паттерны доступа

Структуры данных: B+-деревья и LSM-деревья, read/write amplification

3. Atomic Broadcast и State Machine Replication

Примитив Atomic (Totally Ordered) Broadcast, свойства. АВ как транспорт команд, алгоритм репликации произвольного автомата (RSM), доказательство линейаризуемости. Примеры применения RSM в индустрии.

Большие автоматы, шардирование и транзакции, недетерминизм. Таймауты на клиенте и семантика exactly-once. Недетерминизм. Read-only операции. Параллелизм.

Семинар:

Распределенные файловые системы (DFS). Мотивация. Выбор API, Append/Write.

Дизайн локальной файловой системы: абстракция блочного устройства, слой данных (блоки файлов) и метаданных (namespace, inode-ы).

Разделение DFS на Meta store и Chunk store. Иммутабельность чанков, выбор API для Chunk store, выражение операций DFS через Meta/Chunk store. Реализация Meta store: один узел -> RSM. Реализация chunk store, eventual consistency.

Сравнение с GFS. Операция перезаписи, лизы и primary чанков, гарантии согласованности данных. Гарантии атомарности при пересечении границы чанков.

LSM over DFS.

4. Распределенный консенсус

Atomic Broadcast как Reliable Broadcast + Consensus, эквивалентность задач АВ и Consensus.

Невозможность консенсуса в асинхронной системе: 1) граница $n > 2f$ 2) теорема FLP о невозможности консенсуса в асинхронной сети со сбоями для детерминированных процессов. Практические следствия.

Семинар:

Асинхронность в программировании: корутины / фиберы / фьючи

5. Алгоритм Single Decree Paxos

История алгоритма: статья Part Time Parliament, греки и репликация, статья Paxos Made Simple. Общая идея алгоритма, протокол, разбор сценариев. Интуиция для фазы Prepare. Понятие выбора, контрпример для большинства акцепторов с одним значением. Доказательство корректности. FLP и сценарий лайвлока – dueling proposers. Извлечение выбранного значения. Оптимизации.

Семинар:

Программная симуляция распределенной системы.

6. Алгоритм Multi-Paxos

Эффективная реализация Atomic Broadcast, общая схема RSM, репликация лога команд. Примеры применения RSM в реальных системах.

Multi-Paxos: независимые инстансы консенсуса в для каждого слота лога. Выбор лидера, алгоритм Лэмпорта и его недостатки, ортогональность выбора лидера и репликации, ситуация двух лидеров. Пайплайнинг, ускорение протокола на быстром пути до одного RTT. Масштабирование фазы Prepare на суффикс лога, понятие эпохи. Правила коммита команды лога.

Семинар:

Алгоритм RAFT. Роли, термы, фазы выбора лидера и репликации. Алгоритм выбора лидера в терме. Нетривиальные сценарии, правила коммита и правила голосования. Сравнение RAFT и Multi-Paxos. Разбор промышленной реализации RAFT.

7. Paxos Made Live

Применение Multi-Paxos в промышленной системе. Выбор числа реплик. Расположение реплик, failure domains. Задача переконфигурации: наивный подход, служебная команда в протоколе Multi-Paxos, α -метод. Read-only операции, кворумное подтверждение, использование часов и leader leases. Групповой коммит. Компактификация лога и снимки состояния. Снимки: персистентность, CoW и fork, fuzzy snapshots в ZK. Устройство лога команд: сегментирование, чексуммы, преаллокация и fsync.

Семинар:

Консенсус как сервис: Google Chubby и Apache ZooKeeper. Применение ZK в промышленных распределенных системах.

Crash consistency и файловые системы

8. Распределенные транзакции.

Транзакции, ACID, изоляция транзакций, сериализуемость. consistency models и гарантии изоляции ANSI, аномалии.

Конфликтная сериализуемость, механизм двухфазных блокировок.

Мультиверсионность и снимоты, изоляция снимотов, Google Percolator - протокол транзакций поверх BigTable.

Двухфазный коммит(2PC) в Google Spanner, применение TrueTime.

Детерминированные транзакции (Calvin)

9. Формальные методы для верификации распределенных систем

Проблемы дизайна и верификации распределенных систем, стандартные подходы к верификации. Формальная спецификация и explicit model checking. Граф конфигураций для распределенной системы в асинхронной модели. Масштаб моделей для практической проверки и почему такого масштаба достаточно. Свойства safety и liveness для распределенных систем, линейная темпоральная логика (LTL), выражение типичных свойств для конкурентных / распределенных алгоритмов / объектов в LTL. Язык TLA+.

Разбор спецификаций TLA+ для Single Decree Paxos, RAFT. Техники моделирования распределенных алгоритмов и систем на TLA+.

Язык PlusCal для моделирования многопоточных алгоритмов. Трансляция PlusCal в TLA+.

Fault Injection на примере фреймворка Jepsen: инструменты для внедрения сбоев сети / времени, различные сценарии партишенов, тестирование линеаризуемости.

10. Византийские отказы

Византийская модель сбоев. Причины и примеры византийского поведения. Почему промышленные системы не учитывают византийские сбои. Аутентификация и цифровые подписи. Граница $n > 3f$ для задачи консенсуса, переход через границу в византийской модели и модели с отказами узлов. Рандомизированный алгоритм Ben-Or, кворумы для византийских алгоритмов.

Семинар:

Криптографические инструменты: хэш-функции, цифровые подписи, сертификаты, TLS

11. Алгоритм Practical Byzantine Fault Tolerance (PBFT)

Репликация автомата в византийском окружении. Public Key Infrastructure. Получение ответа от византийской системы. Варианты византийского поведения primary. Фазы Pre-Prepare и Prepare. Rotating primary, протокол перехода через эпоху, кворумные сертификаты. Локальное знание и фаза Commit. Снимки состояния автомата. Цифровые подписи и коды аутентификации сообщений.

Семинар:

Разбор реализации PBFT.

12. Bitcoin. Алгоритм HotStuff

Общая схема электронных денег, граф транзакций и цифровые подписи. Проблема double spending и лог транзакций, задача репликация лога в византийском окружении. Децентрализация и публичность, анонимность и псевдонимность, динамический набор реплик и публичные ключи в качестве адресов. Блокчейн, блоки, транзакции, gossiping. Децентрализованная лотерея – PoW, форки, стабилизация, атака 51%, finality. Мотивация майнеров и эмиссия монет. Блокчейн через линзы классических алгоритмов репликации, сравнение с PBFT.

Семинар:

Selfish mining, шардирование и транзакции между блокчейнами (atomic swaps), анонимность и доказательства с нулевым разглашением

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

Учебная аудитория, оснащенная компьютерами для каждого студента.

6.Перечень рекомендуемой литературы

Основная литература
Не предусмотрено

Дополнительная литература
Не предусмотрено

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

T. Chandra, S. Toueg / Unreliable Failure Detectors for Reliable Distributed Systems / 1996 / <https://dl.acm.org/doi/10.1145/226643.226647>

M. Fischer, N. Lynch, M. Paterson / Impossibility of Distributed Consensus with One Faulty Process / 1985 / <https://groups.csail.mit.edu/tds/papers/Lynch/jacm85.pdf>

L. Lamport / The Part-Time Parliament / 1998 / <https://lamport.azurewebsites.net/pubs/lamport-paxos.pdf>

D. Ongaro and J. Ousterhout / In Search of an Understandable Consensus Algorithm / 2014 / <https://raft.github.io/raft.pdf>

M. Castro, B. Liskov / Practical Byzantine Fault Tolerance / 1999 / <http://pmg.csail.mit.edu/papers/osdi99.pdf>

Nakamoto / Bitcoin: A Peer-to-Peer Electronic Cash System / 2008 / <https://bitcoin.org/bitcoin.pdf>

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

ПО для разработки и отладки программ на языке программирования C++.

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

Лабораторные работы и выполнение домашнего задания является одной общей работой студента. На занятиях особое внимание обращается на современные способы разработки программных продуктов. Необходимо требовать от студентов качественного и понятного программного кода наряду и в равной степени с правильной работой программы. На занятиях стоит постоянно разбирать ошибки, которые допускали студенты в своих программах, а так же логические ошибки.

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

по направлению: Информатика и вычислительная техника

профиль подготовки: Физтех-школа Прикладной Математики и Информатики
кафедра алгоритмов и технологий программирования

курс: 3

квалификация: бакалавр

Семестр, формы промежуточной аттестации: 5 (осенний) - Дифференцированный зачет

Разработчик: Р.Г. Липовский, старший преподаватель

1. Компетенции, формируемые в процессе изучения дисциплины

Код и наименование компетенции	Индикаторы достижения компетенции
ОПК-4 Способен осуществлять сбор и обработку научно-технической и (или) технологической информации для решения фундаментальных и прикладных задач	ОПК-4.1 Владеет методами научного поиска и интеллектуального анализа информации при решении задач профессиональной деятельности
	ОПК-4.2 Знает основные источники научно-технической и (или) технологической информации в области профессиональной деятельности
	ОПК-4.4 Владеет навыками работы с компьютером и компьютерными сетями с целью получения, хранения и обработки научной (технической, технологической) информации
	ОПК-4.3 Умеет составлять аннотации, рефераты, библиографические перечни и обзоры информации в области своей профессиональной деятельности
ОПК-5 Способен участвовать в проведении фундаментальных и прикладных исследований и разработок, самостоятельно осваивать новые теоретические, в том числе, математические методы исследований и работать на современной экспериментальной научно-исследовательской, измерительно-аналитической и технологической аппаратуре)	ОПК-5.1 Способен решать поставленные задачи в области теоретических и экспериментальных исследований и разработок
	ОПК-5.2 Обладает способностью к освоению новых знаний на основе изучения литературы, научных статей и других источников
	ОПК-5.3 Способен к профессиональной эксплуатации современной экспериментальной научно-исследовательской (измерительно-аналитической и технологической) аппаратуры
ПК-1 Способен ставить, формализовывать и решать задачи, в том числе разрабатывать и исследовать математические модели изучаемых явлений и процессов, системно анализировать научные проблемы, получать новые научные результаты	ПК-1.1 Способен находить, анализировать и обобщать информацию об актуальных результатах исследований в рамках тематической области своей профессиональной деятельности
	ПК-1.2 Способен выдвигать гипотезы, строить математические модели для описания изучаемых явлений и процессов, оценить качество разработанной модели
	ПК-1.3 Способен применять теоретические и (или) экспериментальные методы исследований к конкретной научной задаче и интерпретировать полученные результаты
ПК-2 Способен самостоятельно или в качестве члена (руководителя) малого коллектива организовывать и проводить научные исследования и их апробацию	ПК-2.1 Знает принципы построения научной работы, методы сбора и анализа полученного материала, способы аргументации
	ПК-2.2 Способен планировать и проводить научные исследования самостоятельно или в качестве члена (руководителя) малого научного коллектива
	ПК-2.3 Способен проводить апробацию результатов научно-исследовательской работы посредством публикации научных статей и участия в конференциях

2. Показатели оценивания компетенций

В результате изучения дисциплины «Теория отказоустойчивых распределенных систем» обучающийся должен:

знать:

- Алгоритмы, решающие задачи вычислительной геометрии.
- Методики распараллеливания алгоритмов, способы синхронизации потоков, разделения доступа к данным и контроля исполнения подзадач.
- Оценки сложности стандартных алгоритмов.

уметь:

- Реализовывать алгоритмы, решающие задачи вычислительной геометрии.
- Реализовывать параллельные алгоритмы различной, выполнять синхронизацию потоков и доступа к данным.

владеть:

- Средствами стандартной библиотеки C++ для создания многопоточных приложений.
- Методами декомпозиции задач в области информационных технологий и построения единого решения с использованием изученных алгоритмов.

3. Перечень типовых (примерных) вопросов, заданий, тем для подготовки к текущему контролю

1. Доказательство корректности алгоритма RAFT.

Докажите Leader Completeness Property – пусть в терме t лидер закомитил в лог некоторую команду, тогда эта команда будет присутствовать в логе лидера во всех последующих термах.

2. Написание спецификаций для алгоритмов репликации регистра.

Напишите спецификации для двух алгоритмов репликации регистра для случая 1 писателя.

3. Анализ кода системы Apache ZooKeeper.

- Найдите код автомата, который реплицирует ZooKeeper
- Найдите код, который выполняет сегментирование лога транзакций и преаллокацию сегментов
- Найдите код, который пишет снимок дерева. Как он синхронизирован с конкурирующими апдейтами?

4. Анализ статьи про Google BigTable.

Прочтите и разберите статью Bigtable: A Distributed Storage System for Structured Data.

Некоторые контрольные вопросы по статье:

- Что такое таблет в BigTable?
- Какие данные таблета хранятся в оперативной памяти обслуживающего его таблет-сервера?
- Зачем в LSM нужен WAL? Почему таблет-сервер ведет сразу два журнала?

4. Перечень типовых (примерных) вопросов и тем для проведения промежуточной аттестации обучающихся

1. Модель распределенной системы. Время
2. Репликация, отказоустойчивый атомарный регистр
3. State Machine Replication, Atomic Broadcast и Consensus
4. Невозможность консенсуса
5. Single-Decree Paxos
6. Multi-Paxos, RAFT
7. Paxos Made Live
8. Распределенные транзакции
9. Верификация, формальные методы
10. Византийские сбои
11. Practical Byzantine Fault-Tolerance
12. Bitcoin и блокчейны

Примеры билетов:

1. Модель распределенной системы. Время. Bitcoin и блокчейны.
2. Распределенные транзакции. Репликация, отказоустойчивый атомарный регистр.
3. Bitcoin и блокчейны. Невозможность консенсуса.

Критерии оценивания

отлично

10 всесторонние, систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, свободное и правильное обоснование принятых решений;

9 систематизированные, глубокие знания учебной программы дисциплины и умение уверенно применять их на практике при решении конкретных задач, правильное обоснование принятых решений;

8 глубокие знания учебной программы дисциплины и умение применять их на практике при решении конкретных задач, правильное обоснование принятых решений;

хорошо

7 твердо знает материал, грамотно и по существу излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;

6 знает материал, грамотно излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач некоторые неточности;

5 знает основной материал, грамотно излагает его, умеет применять полученные знания на практике, но допускает в ответе или в решении задач неточности;

удовлетворительно

4 фрагментарный, разрозненный характер знаний, недостаточно правильные формулировки базовых понятий, нарушения логической последовательности в изложении программного материала, но при этом он владеет основными разделами учебной программы, необходимыми для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;

3 характер знаний достаточен для дальнейшего обучения и может применять полученные знания по образцу в стандартной ситуации;

неудовлетворительно

2 не знает большей части основного содержания учебной программы дисциплины, допускает грубые ошибки в формулировках основных понятий дисциплины и не умеет правильно использовать полученные знания при решении типовых практических задач.

1 не знает формулировок основных понятий дисциплины и не умеет использовать полученные знания при решении типовых практических задач.

5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности

Дифференцированный зачет выставляется на основе работы на семинаре и выполнения домашних работ, либо, в случае пересдачи комиссии, выполнения задания и его защиты комиссии.

Оценка за дифференцированный зачет выставляется из соотношения: 30% за теоретическую часть при ответе на билет, и 70% - за практическую по итогам выполненных домашних заданий.

Внимание: неудовлетворительная оценка за каждую из частей является БЛОКИРУЮЩЕЙ, то есть, в случае неудовлетворительной оценки за теоретическую либо практическую часть, итоговая оценка - неудовлетворительно.