

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО
ОБРАЗОВАНИЯ

«Национальный исследовательский ядерный университет «МИФИ»

Обнинский институт атомной энергетики –

филиал федерального государственного автономного образовательного учреждения высшего образования «Национальный
исследовательский ядерный университет «МИФИ»

(ИАТЭ НИЯУ МИФИ)

Одобрено УМС ИАТЭ НИЯУ МИФИ,

Протокол №2-8/2024 От 30.08.2024

**ФОНД
ОЦЕНОЧНЫХ СРЕДСТВ
ПО УЧЕБНОЙ ДИСЦИПЛИНЕ**

Версионирование и коллаборативная разработка с помощью Git

| | |
|------------------------|---|
| Направление подготовки | 01.04.02 – Прикладная математика и информатика |
| Профиль | Математическое моделирование и прикладной анализ данных |
| Квалификация (степень) | магистр |
| Форма | очная |

Обнинск 2024г.

Программа составлена в соответствии с требованиями образовательного стандарта высшего образования национального исследовательского ядерного университета «МИФИ» по направлению подготовки 01.04.02 – Прикладная математика и информатика. (квалификация (степень) магистр).

Программу составил:

_____ С.В. Ермаков, доцент, к.ф.-м.н, доцент

Рецензент:

_____ Г.Е. Деев, доцент, к.ф.-м.н, доцент

Программа рассмотрена на заседании ОИКС

(протокол № 5/7 от «30» июля от 2024 г.)

Руководитель направления подготовки 01.04.02
«Прикладная математика и информатика»

_____ Ермаков С.В.

« ____ » _____ 2024 г.

Область применения

Фонд оценочных средств (ФОС) – является неотъемлемой частью учебно-методического комплекса учебной дисциплины «Версионирование и коллаборативная разработка с помощью Git» и предназначен для контроля и оценки образовательных достижений обучающихся, освоивших программу данной дисциплины.

Цели и задачи фонда оценочных средств

Целью Фонда оценочных средств является установление соответствия уровня подготовки обучающихся требованиям федерального государственного образовательного стандарта.

Для достижения поставленной цели Фондом оценочных средств по дисциплине «Версионирование и коллаборативная разработка с помощью Git» решаются следующие задачи:

- контроль и управление процессом приобретения обучающимися знаний, умений и навыков предусмотренных в рамках данного курса;
- контроль и оценка степени освоения компетенций, предусмотренных в рамках данного курса;
- обеспечение соответствия результатов обучения задачам будущей профессиональной деятельности через совершенствование традиционных и внедрение инновационных методов обучения в образовательный процесс в рамках данного курса.

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

1.1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

В результате освоения ООП магистратуры обучающийся должен овладеть следующими результатами обучения по дисциплине:

| Коды компетенций | Результаты освоения ООП Содержание компетенций* | Перечень планируемых результатов обучения по дисциплине** |
|------------------|--|---|
| ОПК-2 | Способен совершенствовать и реализовывать новые математические методы решения прикладных задач | З-ОПК-2 Знать основные понятия, математические методы решения прикладных задач, принципы математического моделирования и методы верификации. У-ОПК-2 Уметь применять полученную теоретическую базу для решения практических задач В-ОПК-2 Владеть основными математическими методами решения прикладных задач |
| ОПК-1 | Способен решать актуальные задачи фундаментальной и прикладной математики | З-ОПК-1 Знать актуальные задачи фундаментальной и прикладной математики, методы математического моделирования. У-ОПК-1 Уметь использовать методы математического моделирования для решения задач фундаментальной и прикладной математики. В-ОПК-1 Владеть методами математического моделирования и основами их использования. |

1.2. Этапы формирования компетенций в процессе освоения ООП магистратура

Компоненты компетенций, как правило, формируются при изучении нескольких дисциплин, а также в немалой степени в процессе прохождения практик, НИР и во время самостоятельной работы обучающегося. Выполнение и защита ВКР являются видом учебной деятельности, который завершает процесс формирования компетенций.

Место дисциплины и соответствующий этап формирования компетенций в целостном процессе подготовки по образовательной программе можно определить по матрице компетенций.

Этапы формирования компетенции в процессе освоения дисциплины:

- **начальный** этап – на этом этапе формируются знаниевые и инструментальные основы компетенции, осваиваются основные категории, формируются базовые умения. Студент воспроизводит термины, факты, методы, понятия, принципы и правила; решает учебные задачи по образцу;
- **основной** этап – знания, умения, навыки, обеспечивающие формирование компетенции, значительно возрастают, но еще не достигают итоговых значений. На этом этапе студент осваивает аналитические действия с предметными знаниями по дисциплине, способен самостоятельно решать учебные задачи, внося коррективы в алгоритм действий, осуществляя коррекцию в ходе работы, переносит знания и умения на новые условия;
- **завершающий** этап – на этом этапе студент достигает итоговых показателей по

заявленной компетенции, то есть осваивает весь необходимый объем знаний, овладевает всеми умениями и навыками в сфере заявленной компетенции. Он способен использовать эти знания, умения, навыки при решении задач повышенной сложности и в нестандартных условиях.

Этапы формирования компетенций в ходе освоения дисциплины отражаются в тематическом плане (см.п. 4 рабочей программы дисциплины).

1.3. Паспорт фонда оценочных средств по дисциплине

Зачет с оценкой

В билете два теоретических вопроса и один практический.

Теоретические вопросы билета:

1. Как создать новый локальный репозиторий и подключить его к удаленному?
2. Чем отличается команда `git init` от команды `git clone`?
3. Как добавить изменения в индекс и выполнить коммит в локальном репозитории?
4. Как отменить изменения, которые ещё не были закоммичены, и как удалить файл из репозитория, но оставить его на диске?
5. Как просмотреть статус локального репозитория и историю коммитов с помощью команд `git status` и `git log`?
6. Чем команда `git fetch` отличается от `git pull` и когда следует использовать каждую?
7. Как синхронизировать локальный репозиторий с удалённым и отправить изменения с помощью `git push`?
8. Как создать, переключиться и удалить локальные ветки, а также просматривать список всех веток?
9. Как объединить изменения из одной ветки в другую с помощью `git merge`, и чем эта команда отличается от `git rebase`?
10. Как разрешать конфликты при слиянии веток или перебазировании, и в чем заключается разница между этими процессами?
11. Как отменить последний коммит, сохранив изменения в рабочем каталоге, с помощью команды `git reset`?
12. Что такое теги в Git и как их использовать для маркировки важных коммитов или версий?
13. Как использовать команду `git rebase` для упорядочивания и чистки истории коммитов, и как избежать потери данных при этом?
14. Как клонировать чужой репозиторий и создать `pull request` для отправки изменений?
15. Как подключить чужой репозиторий как удалённый в своём локальном репозитории и синхронизировать изменения?
16. Как выполнить форк чужого репозитория на GitHub и потом внести изменения в свой форк?
17. Как просматривать и сравнивать изменения в чужом репозитории с помощью `git diff`?
18. Как настроить несколько удалённых репозиториях в одном проекте и работать с ними?
19. Как вернуться к предыдущей версии репозитория с помощью `git checkout <commit>` или `git reset`?

20. Как использовать команды `git branch`, `git checkout`, и `git merge` для эффективного управления ветвлением в командной разработке?

6.2.2. Контрольная работа № 1

Создайте новый локальный репозиторий.

Создайте и переключитесь на новую ветку с именем `feature-branch`.

Добавьте файл с любым содержимым (например, `file.txt`) и выполните коммит.

Переключитесь обратно на основную ветку (`master` или `main`) и создайте новый файл (например, `readme.md`), затем выполните коммит.

Слияние веток:

Переключитесь обратно на ветку `feature-branch` и внесите изменения в `file.txt`.

Выполните коммит этих изменений.

Вернитесь на основную ветку и выполните слияние (`merge`) с веткой `feature-branch`.

Разрешите любые возможные конфликты, если они возникнут.

Работа с удалённым репозиторием:

Создайте репозиторий на GitHub (или другой платформе).

Подключите его как удалённый репозиторий к своему локальному.

Отправьте локальные изменения (`push`) в удалённый репозиторий.

Создайте `pull request` на GitHub для предложенной вами ветки и объясните, какие изменения были внесены.

История и ревизия:

Используйте команду `git log` для просмотра истории коммитов.

Отмените последний коммит с помощью команды `git reset` (оставив изменения в рабочем каталоге) и выполните новый коммит с другим сообщением.

Работа с тегами:

Создайте тег для последнего коммита с именем `v1.0`.

Просмотрите все теги с помощью `git tag`.

Задание на оценку:

Проверьте, чтобы все шаги были выполнены, и продемонстрируйте использование каждой команды через Git и GitHub.

6.2.2. Контрольная работа № 2

Дана маленькая библиотека, вычисляющая некоторые тригонометрические функции.

Инициализируйте репозиторий в папке `mat_lib`

2) Задайте имя пользователя и `email` глобально

1. Проверьте, что изменения внесены в файл глобальных настроек `.gitconfig`

2. Проверьте, что файл локальных настроек `.git/config` не был изменен

3) Изучите содержимое папки `.git/`

1. Узнайте, на что сейчас указывает `HEAD`

2. Просмотрите файл, на который указывает `HEAD` (в этом пункте есть подвох)

4) Добавьте все файлы в индекс

5) Сделайте первый коммит

1. Просмотрите объект коммита, найдите хэш объекта-дерева корня репозитория

2. Просмотрите объект дерева корня репозитория

3. Проверьте, на что указывает HEAD сейчас
4. Просмотрите файл, на который указывает HEAD
5. Ответьте на вопрос: на что указывает текущая ветка? Для этого просмотрите на объект, на который указывает ветка.
- 6) Выполните файл test.py
 1. Просмотрите статус файлов, чтобы обнаружить, что появились файлы кэша
- 7) Сделайте второй коммит
 1. Просмотрите объект коммита, найдите хэш родительского коммита
 2. Посмотрите, на что сейчас указывает HEAD
 3. Проверьте файл, на который указывает HEAD
 4. Узнайте, на что указывает текущая ветка. Для этого просмотрите на объект, на который указывает ветка.

б) критерии оценивания компетенций (результатов) – правильная работа кода программы, понимание алгоритма метода оптимизации, умение вывести необходимые

в) описание шкалы оценивания:

Каждая задача оценивается по шкале от 0 до 10 баллов.

Контрольная работа считается выполненной успешно при суммарной оценке не ниже 18 баллов.

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

| Форма аттестации | Наименование оценочного средства | Баллы |
|--------------------|----------------------------------|-------|
| Зачет (100 баллов) | Контрольная работа № 1 | 30 |
| | Контрольная работа № 2 | 30 |
| | Ответы на билет | 40 |

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

а) основная учебная литература:

- 1) Scott Chacon, Ben Straub – Pro Git (Second Edition) – 2014 – 488 страниц.
- 2) <https://git-scm.com/doc>

б) дополнительная учебная литература:

-

8. Перечень ресурсов* информационно-телекоммуникационной сети «Интернет» (далее - сеть «Интернет»), необходимых для освоения дисциплины

9. Методические указания для обучающихся по освоению дисциплины

| Вид учебного занятия | Организация деятельности студента |
|----------------------|---|
| Лекция | Написание конспекта лекций: кратко, схематично, последовательно фиксировать основные положения, выводы, формулировки, обобщения; помечать важные мысли, выделять ключевые слова, термины. Проверка терминов, понятий с помощью энциклопедий, словарей, справочников с выписыванием толкований в тетрадь. Обозначить вопросы, термины, материал, который вызывает трудности, пометить и попытаться найти ответ в рекомендуемой литературе. Если самостоятельно не удастся разобраться в материале, необходимо сформулировать вопрос и задать преподавателю на консультации, на практическом занятии. |
| Практические занятия | Проработка рабочей программы, уделяя особое внимание целям и задачам, структуре и содержанию дисциплины. Работа с конспектом лекций, просмотр рекомендуемой литературы. Изучение выбранной предметной области на примерах решения задач семинарских занятий, индивидуальных домашних заданий. |
| Курсовая работа | Не предусмотрена |
| Контрольная работа | Ознакомиться с основной и дополнительной литературой, включая справочные издания, зарубежные источники, основополагающие термины. Попрактиковаться в решении аналогичных домашних задач по всем темам контрольных работ. |
| Лабораторная работа | Не предусмотрена. |
| Подготовка к зачету | При подготовке к зачету необходимо ориентироваться на конспекты лекций и рекомендуемую литературу. |

2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания

Конечными результатами освоения программы дисциплины являются сформированные когнитивные дескрипторы «знать», «уметь», «владеть», расписанные по отдельным компетенциям, которые приведены в п.1.1. Формирование этих дескрипторов происходит в процессе изучения дисциплины по этапам в рамках различного вида учебных занятий и самостоятельной работы.

Выделяются три уровня сформированности компетенций на каждом этапе: пороговый, продвинутый и высокий.

| Уровни | Содержательное описание уровня | Основные признаки выделения уровня | БРС, % освоения | ECTS/Пятибалльная шкала для оценки экзамена/зачета |
|--|---|--|-----------------|--|
| Высокий <i>Все виды компетенций сформированы на высоком уровне в соответствии с целями и задачами дисциплины</i> | Творческая деятельность | <i>Включает нижестоящий уровень.</i> Студент демонстрирует свободное обладание компетенциями, способен применить их в нестандартных ситуациях: показывает умение самостоятельно принимать решение, решать проблему/задачу теоретического или прикладного характера на основе изученных методов, приемов, технологий | 90-100 | A/ Отлично/ Зачтено |
| Продвинутый <i>Все виды компетенций сформированы на продвинутом уровне в соответствии с целями и задачами дисциплины</i> | Применение знаний и умений в более широких контекстах учебной и профессиональной деятельности, нежели по образцу, большей долей самостоятельности и инициативы | <i>Включает нижестоящий уровень.</i> Студент может доказать владение компетенциями: демонстрирует способность собирать, систематизировать, анализировать и грамотно использовать информацию из самостоятельно найденных теоретических источников и иллюстрировать ими теоретические положения или обосновывать практику применения. | 85-89 | B/ Очень хорошо/ Зачтено |
| | | | 75-84 | C/ Хорошо/ Зачтено |
| Пороговый <i>Все виды компетенций сформированы на пороговом уровне</i> | Репродуктивная деятельность | Студент демонстрирует владение компетенциями в стандартных ситуациях: излагает в пределах задач курса теоретически и практически контролируемый материал. | 65-74 | D/Удовлетворительно/ Зачтено |
| | | | 60-64 | E/Посредственно /Зачтено |
| Ниже порогового | Отсутствие признаков порогового уровня: компетенции не сформированы. Студент не в состоянии продемонстрировать обладание компетенциями в стандартных ситуациях. | | 0-59 | Неудовлетворительно/ Зачтено |

Оценивание результатов обучения студентов по дисциплине осуществляется по регламенту текущего контроля и промежуточной аттестации.

Критерии оценивания компетенций на каждом этапе изучения дисциплины для каждого вида оценочного средства и приводятся в п. 4 ФОС. Итоговый уровень сформированности компетенции при изучении дисциплины определяется по таблице. При этом следует понимать, что граница между уровнями для конкретных результатов освоения образовательной программы может смещаться.

| Уровень сформированности компетенции | Текущий контроль | Промежуточная аттестация |
|--------------------------------------|------------------------|--------------------------|
| высокий | высокий | высокий |
| | <i>продвинутый</i> | <i>высокий</i> |
| | <i>высокий</i> | <i>продвинутый</i> |
| продвинутый | <i>пороговый</i> | <i>высокий</i> |
| | <i>высокий</i> | <i>пороговый</i> |
| | продвинутый | продвинутый |
| | <i>продвинутый</i> | <i>пороговый</i> |
| пороговый | <i>пороговый</i> | <i>продвинутый</i> |
| | пороговый | пороговый |
| ниже порогового | пороговый | ниже порогового |
| | ниже порогового | - |

3. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков или опыта деятельности, характеризующих этапы формирования компетенций.

Рейтинговая оценка знаний является интегральным показателем качества теоретических и практических знаний и навыков студентов по дисциплине и складывается из оценок, полученных в ходе текущего контроля и промежуточной аттестации.

Текущий контроль в семестре проводится с целью обеспечения своевременной обратной связи, для коррекции обучения, активизации самостоятельной работы студентов.

Промежуточная аттестация предназначена для объективного подтверждения и оценивания достигнутых результатов обучения после завершения изучения дисциплины.

Текущий контроль осуществляется два раза в семестр: контрольная точка № 1 (КТ № 1) и контрольная точка № 2 (КТ № 2).

Результаты текущего контроля и промежуточной аттестации подводятся по шкале балльно-рейтинговой системы.

| Вид контроля | Этап рейтинговой системы Оценочное спелство | Балл | |
|---|---|--------------|---------------|
| | | Минимум | Максимум |
| Текущий | Контрольная точка № 1 | 17 | 28 |
| | Контрольная работа | 12 | 20 |
| | Работа на семинаре | 2 | 3 |
| | Доклад | 3 | 5 |
| | Контрольная точка № 2 | 18 | 33 |
| | Контрольная работа | 14 | 25 |
| | Работа на семинаре | 4 | 8 |
| Промежуточный | Зачет | 25 | 40 |
| | Теоретический вопрос | 10 | 20 |
| | Теоретический вопрос | 15 | 20 |
| ИТОГО по дисциплине за 2 семестр | | 60 (зачтено) | 100 (зачтено) |

Бонусы и штрафы. По Положению бонус (премиальные баллы) не может превышать 5 баллов. Поощрительные баллы студент получает к своему рейтингу в конце семестра: за активную и регулярную работу на занятиях, за выдающийся доклад, за досрочную сдачу домашнего задания. Штрафы студент получает к своему рейтингу в конце семестра: за несвоевременную сдачу домашнего задания (- 3 балла); за пропуск контрольной работы по неуважительной причине (- 2 балла); за нарушение расписания докладов (-3 балла).

Нормативные сроки проведения текущего контроля:

| Этап рейтинговой системы, оценочное средство | Неделя семестра |
|---|------------------------|
| Контрольная точка № 1 | 7 |
| Контрольная работа | 13 |
| Работа на семинаре | 1-14 |
| Контрольная точка № 2 | 17 |
| Контрольная работа | 16 |
| Работа на семинаре | 8-16 |

Контрольная работа проводится и оценивается лектором.

Работа на семинаре, домашнее задание оцениваются преподавателем, ведущим практические занятия.

Доклад оценивается группой, окончательный балл выставляется преподавателем, ведущим практические занятия.

Зачет ставится преподавателем, ведущим практические занятия по итогам ответа на вопросы зачета и набранных баллов в семестре.

4. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков

ОБНИНСКИЙ ИНСТИТУТ АТОМНОЙ ЭНЕРГЕТИКИ ИАТЭ НИЯУ МИФИ

БИЛЕТ № 1

По курсу «Версионирование и коллаборативная разработка с помощью Git»

1. Чем отличается команда `git init` от команды `git clone`?
2. Как использовать команды `git branch`, `git checkout`, и `git merge` для эффективного управления ветвлением в командной разработке?
3. Создайте новый локальный репозиторий.
Создайте и переключитесь на новую ветку с именем `feature-branch`.
Добавьте файл с любым содержимым (например, `file.txt`) и выполните коммит.
Переключитесь обратно на основную ветку (`master` или `main`) и создайте новый файл (например, `readme.md`), затем выполните коммит.
Слияние веток:
Переключитесь обратно на ветку `feature-branch` и внесите изменения в `file.txt`.
Выполните коммит этих изменений.
Вернитесь на основную ветку и выполните слияние (`merge`) с веткой `feature-branch`.
Разрешите любые возможные конфликты, если они возникнут.
Работа с удалённым репозиторием:
Создайте репозиторий на GitHub (или другой платформе).
Подключите его как удалённый репозиторий к своему локальному.
Отправьте локальные изменения (`push`) в удалённый репозиторий.
Создайте `pull request` на GitHub для предложенной вами ветки и объясните, какие изменения были внесены.
История и ревизия:
Используйте команду `git log` для просмотра истории коммитов.
Отмените последний коммит с помощью команды `git reset` (оставив изменения в рабочем каталоге) и выполните новый коммит с другим сообщением.
Работа с тегами:
Создайте тег для последнего коммита с именем `v1.0`.
Просмотрите все теги с помощью `git tag`.
Задание на оценку:
Проверьте, чтобы все шаги были выполнены, и продемонстрируйте использование каждой команды через Git и GitHub.

ОБНИНСКИЙ ИНСТИТУТ АТОМНОЙ ЭНЕРГЕТИКИ ИАТЭ НИЯУ МИФИ

БИЛЕТ № 2

По курсу «Версионирование и коллаборативная разработка с помощью Git»

1. Как просмотреть статус локального репозитория и историю коммитов с помощью команд `git status` и `git log`?
2. Как просматривать и сравнивать изменения в чужом репозитории с помощью `git diff`?
3. Дана маленькая библиотека, вычисляющая некоторые тригонометрические функции.

Инициализируйте репозиторий в папке `mat_lib`

2) Задайте имя пользователя и email глобально

1. Проверьте, что изменения внесены в файл глобальных настроек `.gitconfig`
2. Проверьте, что файл локальных настроек `.git/config` не был изменен

3) Изучите содержимое папки `.git/`

1. Узнайте, на что сейчас указывает HEAD
2. Просмотрите файл, на который указывает HEAD (в этом пункте есть подвох)

4) Добавьте все файлы в индекс

5) Сделайте первый коммит

1. Просмотрите объект коммита, найдите хэш объекта-дерева корня репозитория
2. Просмотрите объект дерева корня репозитория
3. Проверьте, на что указывает HEAD сейчас
4. Просмотрите файл, на который указывает HEAD

5. Ответьте на вопрос: на что указывает текущая ветка? Для этого просмотрите на объект, на который указывает ветка.

6) Выполните файл `test.py`

1. Просмотрите статус файлов, чтобы обнаружить, что появились файлы кэша

7) Сделайте второй коммит

1. Просмотрите объект коммита, найдите хэш родительского коммита
2. Посмотрите, на что сейчас указывает HEAD
3. Проверьте файл, на который указывает HEAD

4. Узнайте, на что указывает текущая ветка. Для этого просмотрите на объект, на который указывает ветка.

Описание шкалы оценивания:

Билет содержит 2 вопроса по всем разделам дисциплины. БИЛЕТ оценивается в 40 баллов, по 20 баллов за вопрос.

| Оценка | Шкала оценивания | Характеристика | Критерии оценки |
|---------|------------------------|---------------------------|---|
| Отлично | 36-40 баллов (90-100%) | Полный ответ на 2 вопроса | Студент должен: - продемонстрировать глубокое усвоение материала; - исчерпывающе, последовательно, грамотно и логически стройно изложить теоретический материал; - правильно формулировать определения |

| | | | |
|---------------------|--------------------------|--|---|
| Хорошо | 30-35 баллов (75-89%) | Неполный ответ | Студент должен: - продемонстрировать достаточное знание материала; - продемонстрировать знание основных теоретических понятий; - достаточно последовательно, грамотно и логически стройно излагать материал; - продемонстрировать умение ориентироваться в литературе |
| Удовлетворительно | 24-29 баллов (60-74%) | Частичный ответ | Студент должен: - продемонстрировать общее знание материала; - показать общее владение понятийным аппаратом дисциплины; - уметь строить ответ в соответствии со структурой излагаемого вопроса |
| Неудовлетворительно | 0-23 баллов (0-59%) | Неудовлетворительный ответ, отсутствие ответа | Студент демонстрирует: - незнание значительной части материала; - не владение понятийным аппаратом дисциплины; - существенные ошибки при изложении материала - не способен привести корректный пример ни по одному вопросу |

Контрольная работа № 1 оценивается в 20 баллов по 2 балла за каждый вопрос. Контрольная работа № 2 оценивается в 25 баллов по 5 баллов за каждый вопрос.
по первой работе

| Оценка | Шкала | Характеристика | Критерии оценки |
|---------------------|------------------------------------|---|--|
| Отлично | Количество набранных баллов: 20-17 | Полный правильный ответ на все вопросы | Дан полный исчерпывающий ответ на вопросы, все понятия проиллюстрированы корректными примерами |
| Хорошо | Количество набранных баллов: 16-13 | Неполный ответ | Дан достаточно полный исчерпывающий ответ на вопросы, имеются незначительные погрешности |
| Удовлетворительно | Количество набранных баллов: 12-10 | Частичный ответ | Ответ на вопросы носит общий характер, отсутствует детализация важных понятий, имеются несущественные ошибки |
| Неудовлетворительно | Количество набранных баллов: <10 | Неудовлетворительный ответ, отсутствие ответа | Ответ неверный, определения не приведены |

по второй работе

| Оценка | Шкала | Характеристика | Критерии оценки |
|---------------------|------------------------------------|---|--|
| Отлично | Количество набранных баллов: 25-20 | Полный правильный ответ на все вопросы | Дан полный исчерпывающий ответ на вопросы, все понятия проиллюстрированы корректными примерами |
| Хорошо | Количество набранных баллов: 19-16 | Неполный ответ | Дан достаточно полный исчерпывающий ответ на вопросы, имеются незначительные погрешности |
| Удовлетворительно | Количество набранных баллов: 15-12 | Частичный ответ | Ответ на вопросы носит общий характер, отсутствует детализация важных понятий, имеются несущественные ошибки |
| Неудовлетворительно | Количество набранных баллов: <12 | Неудовлетворительный ответ, отсутствие ответа | Ответ неверный, определения не приведены |