

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Лиодинович
Должность: Ректор
Дата подписания: 06.04.2026 13:18:12
Уникальный программный ключ:
5cf0d6f89e80f49a334f6a4ba58e91f3326b9926

Приложение А

(обязательное к рабочей программе дисциплины)

**Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Дагестанский государственный технический университет»**

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

по дисциплине «Введение в программирование на языке Python»

Уровень образования

специалитет

(бакалавриат/магистратура/специалитет)

Специальность

**10.05.03 Информационная безопасность
автоматизированных систем**

(код, наименование специальности)

Специализация

Безопасность открытых информационных систем

(наименование)

Разработчик



подпись

Качаева Г.И.

(ФИО уч. степень, уч. звание)

Фонд оценочных средств обсужден на заседании кафедры ИБиПИ **«15» октября 2025г.**, протокол № 2

Зав. кафедрой



подпись

Качаева Г.И.

(ФИО уч. степень, уч. звание)

г. Махачкала 2025

СОДЕРЖАНИЕ

1. Область применения, цели и задачи фонда оценочных средств	3
2. Описание показателей и критериев оценивания компетенций, формируемых в процессе освоения дисциплины (модуля)	3
2.1. Перечень компетенций с указанием этапов их формирования в процессе освоения ОПОП	4
2.1.2. Этапы формирования компетенций	6
2.2. Показатели уровней сформированности компетенций на этапах их формирования, описание шкал оценивания	8
2.2.1. Показатели уровней сформированности компетенций на этапах их формирования	8
2.2.2. Описание шкал оценивания	10
3. Типовые контрольные задания, иные материалы и методические рекомендации, необходимые для оценки сформированности компетенций в процессе освоения ОПОП	11
3.1. Задания и вопросы для входного контроля	11
3.2. Оценочные средства и критерии сформированности компетенций	11
3.2.1. Аттестационная контрольная работа №1	11
3.2.2. Аттестационная контрольная работа №2	12
3.2.3. Аттестационная контрольная работа №3	12
3.2.4. Список вопросов к зачету	13

1. Область применения, цели и задачи фонда оценочных средств

Фонд оценочных средств (ФОС) является неотъемлемой частью рабочей программы дисциплины «Введение в программирование на языке Python» и предназначен для контроля и оценки образовательных достижений обучающихся (в т.ч. по самостоятельной работе студентов, далее – СРС), освоивших программу данной дисциплины.

Целью фонда оценочных средств является установление соответствия уровня подготовки обучающихся требованиям ФГОС ВО по специальности 10.05.03 Информационная безопасность автоматизированных систем.

Рабочей программой дисциплины «Введение в программирование на языке Python» предусмотрено формирование следующих компетенций:

ОПК-2 Способен применять информационно-коммуникационные технологии, программные средства системного и прикладного назначения, в том числе отечественного производства, для решения задач профессиональной деятельности.

2. Описание показателей и критериев оценивания компетенций, формируемых в процессе освоения дисциплины (модуля)

Описание показателей и критериев оценивания компетенций, формируемых в процессе освоения дисциплины (модуля), и используемые оценочные средства приведены в таблице 1.

Перечень оценочных средств, рекомендуемых для заполнения таблицы 1 (в ФОС не приводится, используется только для заполнения таблицы)

- *Эссе*
- *Устный опрос*
- *Вопросы для проведения зачета*

2.1. Перечень компетенций с указанием этапов их формирования в процессе освоения ОПОП

Таблица 1

Код и наименование формируемой компетенции	Код и наименование индикатора достижения формируемой компетенции	Критерии оценивания	Наименование контролируемых разделов и тем ¹
ОПК-2 Способен применять информационно-коммуникационные технологии, программные средства системного и прикладного назначения, в том числе отечественного производства, для решения задач профессиональной деятельности	ОПК-2.1.1 знает состав, назначение функциональных компонентов и программного обеспечения персонального компьютера	<p>Знать: Архитектуру и состав функциональных компонентов персонального компьютера, их назначение и принципы взаимодействия. Классификацию и назначение программного обеспечения, роль операционной системы и интерпретатора в процессе исполнения программ на языке Python.</p> <p>Уметь: Идентифицировать основные функциональные компоненты ПК, определять их характеристики и влияние на производительность при выполнении вычислительных задач. Инсталлировать, настраивать и обновлять программное обеспечение, необходимое для разработки на Python.</p> <p>Владеть: Навыками работы с операционной системой для управления файловой системой, процессами и ресурсами компьютера при разработке и исполнении программ на Python. Навыками подбора конфигурации программно-аппаратных средств для решения конкретных задач профессиональной деятельности.</p>	№№ 1-17
	ОПК-2.1.2 знает формы и способы представления данных в персональном компьютере	<p>Знать: Архитектуру и состав функциональных компонентов персонального компьютера, их назначение и принципы взаимодействия. Классификацию и назначение программного обеспечения, роль операционной системы и интерпретатора в процессе исполнения программ на языке Python.</p> <p>Уметь: Идентифицировать основные функциональные компоненты ПК, определять их характеристики и влияние на производительность при выполнении вычислительных задач. Инсталлировать, настраивать и обновлять программное</p>	№№ 1-17

¹ Наименования разделов и тем должен соответствовать рабочей программе дисциплины.

		<p>обеспечение, необходимое для разработки на Python. Владеть: Навыками работы с операционной системой для управления файловой системой, процессами и ресурсами компьютера при разработке и исполнении программ на Python. Навыками подбора конфигурации программно-аппаратных средств для решения конкретных задач профессиональной деятельности.</p>	
	<p>ОПК-2.2.1 умеет применять типовые программные средства сервисного назначения и пользоваться сетевыми средствами для обмена данными, в том числе с использованием глобальной информационной сети Интернет</p>	<p>Знать: Назначение и функциональные возможности типовых программных средств сервисного назначения и интегрированных сред разработки для Python. Основные протоколы и принципы организации обмена данными в компьютерных сетях, включая глобальную сеть Интернет. Уметь: Использовать сетевые возможности языка Python (модули для работы с сетью) для обмена данными, например, для получения информации из интернета или взаимодействия с удаленными серверами. Применять интегрированную среду разработки и отладчик для написания, тестирования и анализа кода. Владеть: Навыками поиска, анализа и использования информации из глобальной сети Интернет для решения задач программирования. Навыками применения типовых программных средств для организации эффективного рабочего процесса разработчика</p>	<p>№№ 1-17</p>

2.1.2. Этапы формирования компетенций

Сформированность компетенций по дисциплине «Введение в программирование на языке Python» определяется на следующих этапах:

1. **Этап текущих аттестаций** (Для проведения текущих аттестаций могут быть использованы оценочные средства, указанные в разделе 2)

2. **Этап промежуточных аттестаций** (Для проведения промежуточной аттестации могут быть использованы другие оценочные средства)

Таблица 2

Код и наименование формируемой компетенции	Код и наименование индикатора достижения формируемой компетенции	Этапы формирования компетенции					Этап промежуточной аттестации
		Этап текущих аттестаций				18-20 неделя	
		1-5 неделя	6-10 неделя	11-15 неделя	1-17 неделя		
		Текущая аттестация №1	Текущая аттестация №2	Текущая аттестация №3	СРС		КР/КП
1		2	3	4	5	6	7
ОПК-2 Способен применять информационно-коммуникационные технологии, программные средства системного и прикладного назначения, в том числе отечественного производства, для решения задач профессиональной деятельности	ОПК-2.1.1 знает состав, назначение функциональных компонентов и программного обеспечения персонального компьютера	Контрольная работа №1	Контрольная работа №2	Контрольная работа №3			Вопросы для проведения зачета
	ОПК-2.1.2 знает формы и способы представления данных в персональном компьютере	Контрольная работа №1	Контрольная работа №2	Контрольная работа №3			Вопросы для проведения зачета
	ОПК-2.2.1 умеет применять типовые программные средства сервисного назначения и пользоваться сетевыми	Контрольная работа №1	Контрольная работа №2	Контрольная работа №3			Вопросы для проведения зачета

	средствами для обмена данными, в том числе с использованием глобальной информационной сети Интернет						
--	--	--	--	--	--	--	--

СРС – самостоятельная работа студентов;

КР – курсовая работа;

КП – курсовой проект.

2.2. Показатели уровней сформированности компетенций на этапах их формирования, описание шкал оценивания

2.2.1. Показатели уровней сформированности компетенций на этапах их формирования

Результатом освоения дисциплины «Введение в программирование на языке Python» является установление одного из уровней сформированности компетенций: высокий, повышенный, базовый, низкий.

Таблица 3

Уровень	Универсальные компетенции	Общепрофессиональные/ профессиональные компетенции
Высокий (оценка «отлично», «зачтено»)	Сформированы четкие системные знания и представления по дисциплине. Ответы на вопросы оценочных средств полные и верные. Даны развернутые ответы на дополнительные вопросы. Обучающимся продемонстрирован высокий уровень освоения компетенции	Обучающимся усвоена взаимосвязь основных понятий дисциплины, в том числе для решения профессиональных задач. Ответы на вопросы оценочных средств самостоятельны, исчерпывающие, содержание вопроса/задания оценочного средства раскрыто полно, профессионально, грамотно. Даны ответы на дополнительные вопросы. Обучающимся продемонстрирован высокий уровень освоения компетенции
Повышенный (оценка «хорошо», «зачтено»)	Знания и представления по дисциплине сформированы на повышенном уровне. В ответах на вопросы/задания оценочных средств изложено понимание вопроса, дано достаточно подробное описание ответа, приведены и раскрыты в тезисной форме основные понятия. Ответ отражает полное знание материала, а также наличие, с незначительными пробелами, умений и навыков по изучаемой дисциплине. Допустимы единичные негрубые ошибки. Обучающимся продемонстрирован повышенный уровень освоения компетенции	Сформированы в целом системные знания и представления по дисциплине. Ответы на вопросы оценочных средств полные, грамотные. Продемонстрирован повышенный уровень владения практическими умениями и навыками. Допустимы единичные негрубые ошибки по ходу ответа, в применении умений и навыков
Базовый (оценка «удовлетворительно», «зачтено»)	Ответ отражает теоретические знания основного материала дисциплины в объеме, необходимом для дальнейшего освоения ОПОП. Обучающийся допускает неточности в ответе, но обладает необходимыми знаниями для их	Обучающийся владеет знаниями основного материал на базовом уровне. Ответы на вопросы оценочных средств неполные, допущены существенные ошибки. Продемонстрирован базовый уровень владения

Уровень	Универсальные компетенции	Общепрофессиональные/ профессиональные компетенции
	устранения. Обучающимся продемонстрирован базовый уровень освоения компетенции	практическими умениями и навыками, соответствующий минимально необходимому уровню для решения профессиональных задач
Низкий (оценка «неудовлетворительно», «не зачтено»)	Демонстрирует полное отсутствие теоретических знаний материала дисциплины, отсутствие практических умений и навыков	

Показатели уровней сформированности компетенций могут быть изменены, дополнены и адаптированы к конкретной рабочей программе дисциплины.

2.2.2. Описание шкал оценивания

В ФГБОУ ВО «ДГТУ» внедрена модульно-рейтинговая система оценки учебной деятельности студентов. В соответствии с этой системой применяются пятибалльная, двадцатибалльная и стобальная шкалы знаний, умений, навыков.

Шкалы оценивания			Критерии оценивания
пятибалльная	двадцатибалльная	стобальная	
«Отлично» - 5 баллов	«Отлично» - 18-20 баллов	«Отлично» - 85 – 100 баллов	Показывает высокий уровень сформированности компетенций, т.е.: <ul style="list-style-type: none"> – продемонстрирует глубокое и прочное усвоение материала; – исчерпывающе, четко, последовательно, грамотно и логически стройно излагает теоретический материал; – правильно формирует определения; – демонстрирует умения самостоятельной работы с нормативно-правовой литературой; – умеет делать выводы по излагаемому материалу.
«Хорошо» - 4 баллов	«Хорошо» - 15 - 17 баллов	«Хорошо» - 70 - 84 баллов	Показывает достаточный уровень сформированности компетенций, т.е.: <ul style="list-style-type: none"> – демонстрирует достаточно полное знание материала, основных теоретических положений; – достаточно последовательно, грамотно логически стройно излагает материал; – демонстрирует умения ориентироваться в нормальной литературе; – умеет делать достаточно обоснованные выводы по излагаемому материалу.
«Удовлетворительно» - 3 баллов	«Удовлетворительно» - 12 - 14 баллов	«Удовлетворительно» - 56 – 69 баллов	Показывает пороговый уровень сформированности компетенций, т.е.: <ul style="list-style-type: none"> – демонстрирует общее знание изучаемого материала; – испытывает серьезные затруднения при ответах на дополнительные вопросы; – знает основную рекомендуемую литературу; – умеет строить ответ в соответствии со структурой излагаемого материала.
«Неудовлетворительно» - 2 баллов	«Неудовлетворительно» - 1-11 баллов	«Неудовлетворительно» - 1-55 баллов	Ставится в случае: <ul style="list-style-type: none"> – незнания значительной части программного материала; – не владения понятийным аппаратом дисциплины; – допущения существенных ошибок при изложении учебного материала; – неумение строить ответ в соответствии со структурой излагаемого вопроса; – неумение делать выводы по излагаемому материалу.

3. Типовые контрольные задания, иные материалы и методические рекомендации, необходимые для оценки сформированности компетенций в процессе освоения ОПОП

3.1. Задания и вопросы для входного контроля

1. Что такое алгоритм? Перечислите основные свойства алгоритма.
2. Дайте определение переменной. Какие типы данных могут храниться в переменных?
3. Опишите назначение и состав функциональных компонентов персонального компьютера (процессор, память, устройства ввода/вывода).
4. Какие системы счисления используются в вычислительной технике? Как перевести число из десятичной системы в двоичную?
5. Чем отличается компиляция от интерпретации? Приведите примеры языков каждого типа.
6. Что такое операционная система? Назовите её основные функции (управление памятью, процессами, файловой системой).
7. Какие способы представления текстовой информации в компьютере вы знаете? Что такое код символа?
8. Что такое компьютерная сеть? Для чего используется глобальная сеть Интернет?
9. Назовите известные вам интегрированные среды разработки (IDE) и их назначение.
10. Что такое "выражение" в программировании? Приведите пример арифметического и логического выражения.

3.2. Оценочные средства и критерии сформированности компетенций

3.2.1. Аттестационная контрольная работа №1

1. Перечислите основные встроенные типы данных в языке Python. Для каких задач предназначен тип `bool`?
2. Какими способами можно организовать ввод данных с клавиатуры и вывод на экран в Python? Приведите примеры.
3. Напишите фрагмент кода, который проверяет, является ли введённое число чётным, и выводит соответствующее сообщение.
4. Объясните разницу между циклами `while` и `for`. В каких случаях предпочтительнее использовать каждый из них?
5. Создайте список, содержащий числа от 1 до 15. Используя срез, получите новый список из элементов с 5-го по 10-й (включительно).
6. Что такое кортеж? Почему его называют неизменяемым типом данных? В каких ситуациях кортежи предпочтительнее списков?
7. Опишите механизм работы цикла `for` с итерируемыми объектами. Что такое итератор?
8. Напишите код, который запрашивает у пользователя 5 целых чисел, сохраняет их в список, а затем вычисляет и выводит среднее арифметическое.
9. Как выполнить распаковку кортежа? Приведите пример, где значения кортежа присваиваются отдельным переменным.
10. Что произойдёт при попытке обращения к несуществующему индексу списка? Как избежать ошибки?
11. Используя генератор списка, создайте список квадратов чисел от 1 до 10.
12. Какие методы списка позволяют добавить элемент в конец (`append`) и вставить элемент по индексу (`insert`)?
13. Чем отличается оператор `is` от оператора `==` при сравнении объектов?
14. Напишите код, который преобразует строку, содержащую числа, разделённые пробелами, в список целых чисел.
15. Объясните принцип работы оператора `in` при проверке наличия элемента в списке или кортеже.

3.2.2. Аттестационная контрольная работа №2

1. Что представляет собой словарь (dict) в Python? Опишите структуру «ключ — значение».
2. Как получить значение из словаря по ключу? Какие существуют безопасные способы (без генерации исключения)?
3. Напишите код для добавления новой пары в словарь или обновления существующей.
4. Перечислите способы перебора элементов словаря (ключи, значения, пары ключ-значение).
5. Что такое множество (set)? Какие операции над множествами реализованы в Python (объединение, пересечение, разность)?
6. Как проверить, является ли элемент частью множества? Какова временная сложность этой операции?
7. В чём отличие set от frozenset?
8. Какие структуры данных предоставляет модуль collections? Для чего используются namedtuple и defaultdict?
9. Приведите пример использования Counter для подсчёта частоты символов в строке.
10. Объясните назначение и преимущества deque по сравнению со списком при операциях добавления/удаления с обоих концов.
11. Напишите код, который создаёт словарь, где ключами являются слова из предложения, а значениями — количество их вхождений.
12. Как удалить дубликаты из списка, сохранив порядок элементов? (Используйте множество и вспомогательный список).
13. Выполните объединение двух словарей dict1 = {'a': 1, 'b': 2} и dict2 = {'b': 3, 'c': 4} с сохранением значений из второго для совпадающих ключей.
14. Как преобразовать список кортежей вида [('key1', 10), ('key2', 20)] в словарь?
15. Что произойдёт при попытке использовать список в качестве ключа словаря? Объясните причину.

3.2.3. Аттестационная контрольная работа №3

1. Опишите синтаксис определения функции в Python. Что такое параметры и аргументы?
2. Чем отличаются позиционные аргументы от именованных? Для чего нужны аргументы по умолчанию?
3. Что такое возвращаемое значение функции? Может ли функция возвращать несколько значений?
4. Объясните понятие области видимости переменных. В чём разница между локальной и глобальной переменной?
5. Приведите примеры встроенных функций высшего порядка (map, filter). Как они работают?
6. Для чего используются lambda-функции? Напишите lambda, которая возвращает квадрат числа.
7. Перечислите основные методы строк: разделение (split), соединение (join), замена (replace), изменение регистра.
8. Какие существуют способы форматирования строк в Python (f-строки, метод format, оператор %)? Приведите примеры.
9. Что такое кодировка символов? Почему при работе с файлами важно указывать кодировку (например, utf-8)?
10. Как открыть файл для чтения в Python? Какие режимы открытия файлов вы знаете (r, w, a, rb, wb)?
11. Напишите код, который построчно читает текстовый файл и выводит каждую строку с её номером.
12. В чём преимущество использования менеджера контекста with при работе с файлами?
13. Как обработать исключение, возникающее при попытке открыть несуществующий файл? Приведите пример с try-except.

14. Какие функции модуля `os` позволяют получать список файлов в директории, создавать папки, проверять существование пути?
15. Что такое объект `Path` из модуля `pathlib`? Как с его помощью удобно конструировать пути к файлам?

3.2.4. Список вопросов к зачету

1. История создания и особенности языка Python. Области применения.
2. Установка интерпретатора Python. Работа в интерактивном режиме и выполнение скриптов.
3. Переменные в Python: правила именования, динамическая типизация. Базовые типы данных (`int`, `float`, `str`, `bool`).
4. Операторы ветвления: `if`, `elif`, `else`. Тернарный условный оператор.
5. Циклы `while` и `for`. Операторы `break`, `continue`, `else` в циклах.
6. Списки: способы создания, индексация, срезы. Основные методы списков (`append`, `extend`, `insert`, `remove`, `pop`).
7. Кортежи: неизменяемость, особенности использования, упаковка и распаковка.
8. Итераторы и итерируемые объекты. Протокол итерации. Функции `iter()` и `next()`.
9. Генераторы списков (`list comprehensions`) и аналогичные выражения для множеств и словарей.
10. Словари: создание, доступ к элементам (ключи, значения, пары). Методы `get()`, `keys()`, `values()`, `items()`.
11. Множества: операции над множествами (`union`, `intersection`, `difference`). Методы `add`, `remove`, `discard`.
12. Модуль `collections`: `namedtuple` (именованные кортежи), `defaultdict` (словарь со значением по умолчанию).
13. Модуль `collections`: `Counter` (подсчёт объектов), `deque` (двусторонняя очередь).
14. Функции в Python: определение, вызов. Параметры и аргументы (позиционные, ключевые, произвольное количество).
15. Возврат значений из функции. Оператор `return`. Функции, возвращающие несколько значений.
16. Области видимости переменных: локальная, глобальная, `nonlocal`. Правило LEGB.
17. Лямбда-функции. Функции `map()`, `filter()`, `reduce()`.
18. Функции как объекты первого класса. Замыкания (`closures`).
19. Строки: основные методы (поиск, замена, разбиение, соединение). Проверки символов (`isdigit`, `isalpha` и др.).
20. Форматирование строк: f-строки, метод `format()`, старый стиль с `%`.
21. Кодировки символов: ASCII, Unicode, UTF-8. Работа с кодировками при чтении/записи файлов.
22. Регулярные выражения: базовый синтаксис, модуль `re`. Функции `search`, `match`, `findall`, `sub`.
23. Работа с файлами: открытие (`open`), режимы работы. Чтение (`read`, `readline`, `readlines`) и запись (`write`, `writelines`).
24. Менеджер контекста `with`. Автоматическое закрытие файлов.
25. Обработка исключений: конструкция `try-except-else-finally`. Типы исключений.
26. Модуль `os`: работа с файловой системой (переменные окружения, управление путями, создание/удаление директорий).
27. Модуль `pathlib`: объектно-ориентированный интерфейс для работы с путями.
28. Понятие модуля и пакета. Импорт модулей (`import`, `from ... import ...`). Подключение сторонних библиотек.
29. Стандартная библиотека Python: обзор полезных модулей (`math`, `random`, `datetime`, `json`).
30. Основы стиля программирования: PEP 8, рекомендации по оформлению кода.

Зачеты и экзамены могут быть проведены в письменной форме, а также в письменной форме с устным дополнением ответа. Зачеты служат формой проверки качества выполнения студентами лабораторных работ, усвоения семестрового учебного материала по дисциплине (модулю), практических и семинарских занятий (при отсутствии экзамена по дисциплине).

По итогам зачета, соответствии с модульно – рейтинговой системой университета, выставляются баллы с последующим переходом по шкале баллы – оценки за зачет, выставляемый как по наименованию «зачтено», «не зачтено», так и дифференцированно т.е. с выставлением отметки по схеме – «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно», определяемое решением Ученого совета университета и прописываемого в учебном плане.

Экзамен по дисциплине (модулю) служит для оценки работы студента в течении семестра (года, всего срока обучения и др.) и призван выявить уровень, качество и систематичность полученных им теоретических и практических знаний, приобретения навыков самостоятельной работы, развития творческого мышления, умения синтезировать полученные знания и применять их в решении практических задач. По итогам экзамена, в соответствии с модульно – рейтинговой системой университета выставляются баллы, с последующим переходом по шкале оценок на оценки: «отлично», «хорошо», «удовлетворительно» и «неудовлетворительно», свидетельствующие о приобретенных компетенциях или их отсутствии.

Критерии оценки уровня сформированности компетенций по результатам проведения зачета:

- оценка «зачтено»: обучающийся демонстрирует всестороннее, систематическое и глубокое знание материала, свободно выполняет задания, предусмотренные программой дисциплины, усвоивший основную и дополнительную литературу. Обучающийся выполняет задания, предусмотренные программой дисциплины, на уровне не ниже базового;

- оценка «не зачтено»: обучающийся демонстрирует незнание материала, не выполняет задания, предусмотренные программой дисциплины. Обучающийся не выполняет задания, предусмотренные программой дисциплины, на уровне ниже базового. Дальнейшее освоение ОПОП не возможно без дополнительного изучения материала и подготовки к зачету.

Критерии оценки уровня сформированности компетенций по результатам проведения дифференцированного зачёта (зачета с оценкой) / экзамена:

- оценка «**отлично**»: обучающийся дал полный, развернутый ответ на поставленный вопрос, проявил совокупность осознанных знаний об объекте, доказательно раскрыл основные положения темы. В ответе прослеживается четкая структура, логическая последовательность, отражающая сущность раскрываемых понятий, явлений. Обучающийся подкрепляет теоретический ответ практическими примерами. Ответ сформулирован научным языком, обоснована авторская позиция обучающегося. Могут быть допущены недочеты в определении понятий, исправленные студентом самостоятельно в процессе ответа или с помощью «наводящих» вопросов преподавателя. Обучающимся продемонстрирован высокий уровень владения компетенцией(-ями);

- оценка «**хорошо**»: обучающимся дан полный, развернутый ответ на поставленный вопрос, проявлено умение выделять существенные и несущественные признаки, причинно-следственные связи. Ответ четко структурирован, логичен, но есть недочеты в формулировании понятий, решении задач. При ответах на дополнительные вопросы допущены незначительные ошибки. Обучающимся продемонстрирован повышенный уровень владения компетенцией(-ями);

- оценка «**удовлетворительно**»: обучающимся дан неполный ответ на вопрос, логика и последовательность изложения имеют существенные нарушения. Допущены грубые ошибки при определении сущности раскрываемых понятий, явлений, нарушена логика ответа, не сделаны выводы. Речевое оформление требует коррекции. Обучающийся испытывает затруднение при ответе на дополнительные вопросы. Обучающимся продемонстрирован базовый уровень владения компетенцией(-ями);

- оценки «**неудовлетворительно**»: обучающийся испытывает значительные трудности в ответе на вопрос, допускает существенные ошибки, не владеет терминологией, не знает основных понятий, не может ответить на «наводящие» вопросы преподавателя. Обучающимся продемонстрирован низкий уровень владения компетенцией(-ями).