

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Баламирзоев Назим Ибрагимович
Должность: Ректор
Дата подписания: 24.11.2025 11:35:35
Уникальный программный ключ:
5cf0d6f89e80f49a334f6a4ba58e91f3326b9926

Министерство науки и высшего образования Российской Федерации
ФГБОУ ВО «Дагестанский государственный технический университет»
Кафедра «Прикладная математика и информатика»

ОДОБРЕНО:

Методической комиссией по укрупненной
группе специальностей и направлений
01.00.00–Математика и механика

Председатель МК:


Подпись Т.И.Исабекова
ФИО

«__» ____ 20__ г.

УТВЕРЖДАЮ:

Декан факультета
КТВТиЭ


Подпись Т.А. Рагимова
ФИО

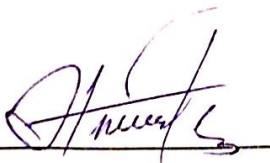
«__» ____ 20__ г.

Фонд оценочных средств

по дисциплине «Введение в программирование на языке
Python»

для контроля знаний обучающихся по направлению подготовки 01.03.02 –
«Прикладная математика и информатика» профиль «Системное программиро-
вание, компьютерные технологии»

Составитель, д.э.н., профессор



Савзиханова С.Э.

Фонд оценочных средств обсужден на заседании кафедры ПМИИ

«__» ____ 20__ г., протокол №__

Зав. кафедрой



Т.И.Исабекова

Фонд оценочных средств является приложением к рабочей программе по дисциплине «Введение в
программирование на языке Python»

Махачкала, 20__ г.

СОДЕРЖАНИЕ

1. Перечень компетенций с указанием этапов их формирования в процессе освоения ООП (Таблицы 1 и 2).....	3
1.1. Перечень компетенций и планируемые результаты.....	3
1.2. Этапы формирования компетенций.....	4
2. Описание показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания.....	5
2.1. Описание показателей оценивания компетенций.....	6
2.2. Описание критериев определения уровня сформированности компетенций.....	7
2.3. Описание шкал оценивания.....	8
2.4. Определение уровня сформированности компетенций в результате изучения дисциплины «Введение в программирование на языке Python»	9
3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения ООП.....	9
3.1. Задания для входного контроля.....	9
3.1.1. Вопросы для входного контроля.....	9
3.2. Задания для текущих аттестаций.....	10
3.2.1. Контрольные вопросы для первой аттестации.....	10
3.2.2. Контрольные вопросы для второй аттестации.....	11
3.2.3. Контрольные вопросы третьей аттестации.....	12
3.3. Задания для промежуточной аттестации (зачета)	13
3.3.1. Контрольные вопросы для проведения экзамена.....	13
3.4. Задания для проверки остаточных знаний.....	15
3.4.1. Вопросы для проверки остаточных знаний.....	15
4. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций.....	17
4.1. Процедура проведения оценочных мероприятий.....	17

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

ООП (Таблицы 1 и 2)

1.1. Перечень компетенций и планируемые результаты

Таблица 1

№	Содержание и код компетенций по ФГОС	В результате изучения дисциплины «Введение в программирование на языке Python» обучающиеся должны:		
		знать	уметь	владеть
1	Способность применять информационно-коммуникационные технологии, программные средства системного и прикладного назначения, в том числе отечественного производства, для решения задач профессиональной деятельности (ОПК-2)	- Основные конструкции и синтаксис языка Python. - Принципы структурного и процедурного программирования. - Состав и назначение базового программного обеспечения для разработки на Python	- Применять переменные, базовые структуры данных и управляющие конструкции для решения задач. - Использовать интегрированные среды разработки (IDE) для создания, отладки и выполнения программ. - Разрабатывать несложные алгоритмы и реализовывать их в виде программ на Python.	- Навыками написания, тестирования и отладки простых программ на языке Python. - Основными методами работы со встроенными структурами данных (списки, словари, кортежи, множества). - Базовыми приемами работы с файлами и строками.

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

1. Этап текущих аттестаций (текущие аттестации 1-3; СРС; КР)
2. Этап промежуточных аттестаций (экзамен – 1 семестр, экзамен – 2 семестр)

1.2. Этапы формирования компетенций

Таблица 2

Код компетенций по ФГОС	Этапы формирования компетенций по дисциплине «Численные методы»						
	СЕМЕСТР						
	I						II
	Этап текущих аттестаций						Этап пром. аттест.
	1-5 нед.	6-10 нед.	11-15 нед.	1-17 нед.		18-20 нед.	
	Текущая аттест. 1 (контр. раб. 1)	Текущая аттест. 2 (контр. раб. 2)	Текущая аттест. 3 (контр. раб. 3)	СРС (творч. отчет)	КР (поясн. зап., ГМ)	Промеж. аттест. (зачет)	
1	2	3	4	5	6	7	8
ОПК-2	+	+	+	+	+	+	

Таблица 2.2

Код компетенций по ФГОС	Этапы формирования компетенций по дисциплине «Численные методы»						
	СЕМЕСТР						
	II						III
	Этап текущих аттестаций						Этап пром. аттест.
	1-5 нед.	6-10 нед.	11-15 нед.	1-17 нед.		18-20 нед.	
	Текущая аттест. 1 (контр. раб. 1)	Текущая аттест. 2 (контр. раб. 2)	Текущая аттест. 3 (контр. раб. 3)	СРС (творч. отчет)	КР (поясн. зап., ГМ)	Промеж. аттест. (зачет)	
1	2	3	4	5	6	7	8
ОПК-2	+	+	+	+	+	+	

СРС – самостоятельная работа студентов;

КР – курсовая работа;

ГМ – графический материал;

Знак «+» соответствует формированию компетенции.

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

В рамках текущих аттестаций (таблица 1) оценка уровня сформированности компетенций проводится в ходе выполнения курсовых работ и проектов, а также на занятиях:

- ~ лабораторного типа посредством экспресс- опроса обучаемых, в том числе по темам и разделам, вынесенных для самостоятельного изучения;
- ~ семинарского типа путем собеседования;
- ~ практического типа методами устного опроса или проведения письменных контрольных работ;

Оценка сформированности компетенций в рамках промежуточной аттестации проводится по билетам для экзамена. Они включают в себя вопросы для оценки знаний, умений и навыков, т.е. задания:

репродуктивного уровня, позволяющие оценивать и диагностировать знание фактического материала (базовые понятия, алгоритмы, факты) и умения правильно использовать специальные термины и понятия, узнавание объектов изучения в рамках определенного раздела дисциплины (модуля);

реконструктивного уровня, позволяющие оценивать и диагностировать умения синтезировать, анализировать, обобщать фактический и теоретический материал с формулированием конкретных выводов, установлением причинно-следственных связей;

творческого уровня, позволяющие оценивать и диагностировать умения, интегрировать знания различных областей, аргументировать собственную точку зрения.

В ходе проведения текущей и промежуточной аттестации оцениваются:

- ~ полнота и содержательность ответа;
- ~ умение привести примеры
- ~ умение отстаивать свою позицию в ходе защиты творческого отчета по самостоятельной работе;
- ~ умение пользоваться дополнительной литературой и современными технологиями обучения (в т.ч. сетевых информационных технологий) при подготовке к занятиям;
- ~ умение применять нормативно-правовые акты при подготовке к занятиям и выполнению индивидуальных занятий;
- ~ соответствие представленной в ответах информации материалам лекций, учебной литературы, интернет- ресурсам и другим источникам информации.

В ходе проведения оценки сформированности компетенций рекомендуются применение современных компьютерных технологий и виртуальных форм опроса в интерактивном режиме.

2.1. Описание показателей оценивания компетенций

Таблица 3.

Оценка «неудовлетворительно» (не зачтено) или отсутствие сформированности компетен- ции	Оценка «удовлетворительно» (зачтено) или низкой уровень освоения компетен- ции	Оценка «хорошо» (зачтено) или повышенный уровень освоения компетенции	Оценка «отлично» (зачтено) или высо- кий уровень освоения компетенции
Неспособность обучающегося самостоятельно продемонстрировать наличие знаний при решении заданий, которые были представлены преподавателем вместе с образцом их решения, отсутствие самостоятельности в применении умения к использованию методов освоения учебной дисциплины и неспособность самостоятельно проявить навыки повторения решения поставленной задачи по стандартному образцу свидетельствуют об отсутствии сформированности компетенции. Отсутствие подтверждения наличия сформированности компетенции свидетельствует об отсутствии сформированности освоения учебных результатов освоения учебной дисциплины. Уровень освоения дисциплины, при котором у обучающегося не сформировано более 50% компетенций. Если же учебная дисциплина выступает в качестве итогового этапа формирования компетенций (чаще всего это дисциплины профессионального цикла) оценка «неудовлетворительно» должна быть выставлена при отсутствии сформированности хотя бы одной компетенции.	Если обучаемый демонстрирует самостоятельность в применении знаний, умений и навыков к решению учебных заданий в полном соответствии с образцом, данным преподавателем, по заданиям, решение которых было показано преподавателем, следует считать, что компетенция сформирована, но ее уровень недостаточно высок. Поскольку выявлено наличие сформированной компетенции, ее следует оценивать положительно, но на низком уровне. При наличии более 50% сформированных компетенций по дисциплинам, имеющим возможность до-формирования компетенций на последующих этапах обучения. Для дисциплин итогового формирования компетенций естественно выставлять оценку «удовлетворительно», если сформированы все компетенции и более 60% дисциплин профессионального цикла «удовлетворительно».	Способность обучающегося продемонстрировать самостоятельное применение знаний, умений и навыков при решении заданий, аналогичных тем, которые представлял преподаватель при потенциальном формировании компетенции, подтверждает наличие сформированной компетенции, причем на более высоком уровне. Наличие сформированной компетенции на повышенном уровне самостоятельности со стороны обучающегося при ее практической демонстрации в ходе решения аналогичных заданий следует оценивать как положительное и устойчиво закрепленное в практическом навыке. Для определения уровня освоения промежуточной дисциплины на оценку «хорошо» обучающийся должен продемонстрировать наличие 80% сформированных компетенций, из которых не менее 1/3 оценены отметкой «хорошо». Оценивание итоговой дисциплины на «хорошо» обуславливается наличием у обучающегося всех сформированных компетенций причин общепрофессиональных компетенции по учебной дисциплине должны быть сформированы не менее чем на 60% на повышенном уровне, то есть с оценкой «хорошо».	Обучаемый демонстрирует способность к полной самостоятельности (допускаются консультации с преподавателем по сопутствующим вопросам) в выборе способа решения неизвестных или нестандартных заданий в рамках учебной дисциплины с использованием знаний, умений и навыков, полученных как в ходе освоения данной учебной дисциплины, так и смежных дисциплин, следует считать компетенцию сформированной на высоком уровне. Присутствие сформированной компетенции на высоком уровне, способность к ее дальнейшему саморазвитию и высокой адаптивности практического применения к изменяющимся условиям профессиональной задачи. Оценка «отлично» по дисциплине с промежуточным освоением компетенций, может быть выставлена при 100% подтверждении наличия компетенций, либо при 90% сформированных компетенций, из которых не менее 2/3 оценены отметкой «хорошо». В случае оценивания уровня освоения дисциплины с итоговым формированием компетенций оценка «отлично» может быть выставлена при подтверждении 100% наличия сформированной компетенции у обучающегося, выполнены требования к получению оценки «хорошо» и освоены на «отлично» не менее 50% общепрофессиональных компетенций.

2.2. Описание критериев определения уровня сформированности компетенций

Таблица 4

Уровни сформированности компетенций	Критерии определения уровня сформированности	Компетенции, формируемые в результате освоения дисциплины ООП
		Общепрофессиональные компетенции (ОПК)
		ОПК-2
Пороговый уровень	Компетенция сформирована	+
	Демонстрируется недостаточный уровень самостоятельности навыка	+
	Обладает качеством репродукции	+
Достаточный уровень	Компетенция сформирована	+
	Демонстрируется достаточный уровень самостоятельности устойчивого практического навыка	+
	Обладает качеством реконструкции	+
Высокий уровень	Компетенция сформирована	+
	Демонстрируется высокий уровень самостоятельности, высокая адаптивность практического навыка	+
	Обладает творческим качеством	+

2.3. Описание шкал оценивания

В Дагестанском государственном техническом университете внедрена модульно-рейтинговая система оценки учебной деятельности студентов. В соответствии с этой системой применяются пятибалльная, двадцатибалльная и стобалльная шкалы знаний, умений, навыков.

Шкалы оценивания			Критерии оценивания
пятибалльная	двадцатибалльная	стобалльная	
«Отлично» - 5 баллов	«Отлично» - 18-20 баллов	«Отлично» - 85 баллов	Показывает высокий уровень сформированности компетенций, т.е.: - продemonстрирует 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.: - 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.: - 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» - 1-56 баллов	Ставится в случае: - 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.

2.4. Определение уровня сформированности компетенций в результате изучения дисциплины

«Введение в программирование на языке Python»

Таблица 6

№	Код компетенций по ФГОС	Уровни сформированности компетенций		
		Пороговый	Достаточный	Высокий
1	2	3	4	5
1.	ОПК-2	Знает • принципы работы программного обеспечения для разработки на Python; • стандартные и нестандартные задачи программирования; • различия между компилируемыми и интерпретируемыми языками программирования: слабо (на пороговом уровне, или на «удовлетворительно»). Умеет решать типовые задачи программирования по образцу слабо. Владет базовыми навыками написания, тестирования и отладки простых программ. слабо.	Знает • принципы работы программного обеспечения для разработки на Python; • стандартные и нестандартные задачи программирования; • различия между компилируемыми и интерпретируемыми языками программирования: на достаточном уровне (на «хорошо»). Умеет решать типовые задачи программирования по образцу на достаточном уровне. Владет базовыми навыками написания, тестирования и отладки простых программ. на достаточном уровне.	Знает • принципы работы программного обеспечения для разработки на Python; • стандартные и нестандартные задачи программирования; • различия между компилируемыми и интерпретируемыми языками программирования: полноценно (на высоком уровне, на «отлично»). Умеет решать типовые задачи программирования по образцу полноценно. Владет базовыми навыками написания, тестирования и отладки простых программ. полноценно.

3. Типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующие этапы формирования компетенций в процессе освоения ООП.

3.1. Задания для входного контроля

3.1.1. Вопросы для входного контроля

- Архитектура компьютера**
 - Перечислите основные компоненты персонального компьютера и их назначение
 - Что такое оперативная память и постоянное хранилище? В чем их отличие?
 - Какие устройства относятся к устройствам ввода/вывода информации?
- Программное обеспечение**
 - Что такое операционная система? Назовите основные функции ОС
 - Какие виды программного обеспечения вы знаете? Приведите примеры
 - Что такое драйвер устройства и для чего он нужен?
- Файловая система**

- Что такое файл и каталог? Какие типы файлов вы знаете?
 - Что такое путь к файлу? Приведите примеры абсолютного и относительного пути
 - Какие операции можно выполнять с файлами и каталогами?
4. **Понятие алгоритма**
- Дайте определение алгоритма. Перечислите основные свойства алгоритмов
 - Какие способы описания алгоритмов вам известны?
 - Что такое блок-схема? Изобразите основные элементы блок-схем
5. **Базовые алгоритмические конструкции**
- Опишите алгоритм решения линейного уравнения $ax + b = 0$
 - Составьте алгоритм нахождения максимального из трех чисел
 - Напишите алгоритм вычисления факториала числа
6. **Системы счисления**
- Переведите число 47 из десятичной системы в двоичную
 - Переведите число 101101_2 из двоичной системы в десятичную
 - Что такое шестнадцатеричная система счисления? Где она применяется?
7. **Единицы измерения информации**
- Что такое бит и байт? Как связаны эти единицы измерения?
 - Сколько байт в 1 килобайте? Сколько бит в 1 мегабайте?
 - Оцените объем текстового документа, содержащего 1500 символов
8. **Базовые логические операции**
- Определите результат логического выражения: $(\text{True AND False}) \text{ OR True}$
 - Составьте таблицу истинности для операции XOR (исключающее ИЛИ)
 - Решите логическую задачу: "Если идет дождь, то я беру зонт. Сегодня я взял зонт. Значит ли это, что идет дождь?"
9. **Алгоритмическое мышление**
- Определите результат выполнения последовательности действий:
text
 $x = 10$
 $y = 5$
 $z = x + y$
 $x = z - y$
 $y = x * 2$
Какие значения будут иметь переменные x , y , z ?
10. **Программные средства**
- Какие офисные программы вы знаете? Для чего они используются?
 - Что такое архиватор? Для чего используется архивация файлов?
 - Какие браузеры вы знаете? Что такое интернет-браузер?
11. **Безопасность работы**
- Что такое компьютерный вирус? Какие меры защиты от вирусов вы знаете?
 - Для чего нужны лицензионные программы?
 - Какие правила безопасной работы в интернете вы можете назвать?
12. **Решение задачи**
- Составьте алгоритм вычисления площади прямоугольника по заданным сторонам
 - Опишите алгоритм определения, является ли число четным или нечетным
 - Составьте блок-схему для нахождения суммы чисел от 1 до N

3.2. Задания для текущих аттестаций

3.2.1. Контрольные вопросы для первой аттестации

1. Напишите программу, которая выводит ваше имя и фамилию в одной строке.
2. Напишите программу, которая считывает два целых числа и выводит их сумму.
3. Напишите программу, которая считывает 3 строку и выводит их.

4. Напишите программу, которая выводит фразу «Курс Основы программирования начался» три раза подряд, разделяя фразы тремя символами "****" (без пробелов).
5. Чему равно значение переменной 'x' после выполнения кода?
``x = 15 % 4 + 7 // 2``
6. Напишите программу, которая считывает имя пользователя и приветствует его, выводя фразу «Привет, [имя]!».
7. Напишите программу, которая считывает три строки и выводит их в обратном порядке, каждую с новой строки.
8. Исправьте ошибки в коде, чтобы программа выводила число 10:
``num = input() print(num + 2)``
9. Напишите программу, которая выводит буквы 'P', 'y', 't', 'h', 'o', 'n' в одну строку, разделяя их символом точки '.'.
10. Чему равно значение выражения ``2 ** 3 ** 0``?
11. Напишите программу, которая выводит название вашего любимого фильма.
12. Напишите программу, которая считывает целое число и выводит предыдущее и следующее за ним числа в формате: «Следующее за числом [число] число: [следующее]», «Для числа [число] предыдущее число: [предыдущее]».
13. Напишите программу, которая считывает слово и выводит его 5 раз подряд без пробелов.
14. Используя только один вызов ``print()``, выведите значения переменных 'a', 'b' и 'c' так, чтобы между ними были знаки «плюс» и «равно». Пример: ``2+2=4``.
15. Чему равно значение переменной 'res' после выполнения кода?
``res = 10 // 3 * 2 + 5``
16. Напишите программу, которая выводит вашу дату рождения в формате ДД.ММ.ГГГГ.
17. Напишите программу, которая считывает два числа и выводит их произведение.
18. Напишите программу, которая считывает строку и выводит её длину (количество символов). Используйте функцию ``len()``.
19. Напишите программу, которая выводит числа от 1 до 3, каждое на новой строке, используя три вызова ``print()``.
20. Чему равно значение выражения ``-5 // 2``?

3.2.2. Контрольные вопросы для второй аттестации

1. Напишите программу, которая определяет, оканчивается ли год с данным номером на два нуля. Если год оканчивается, то выведите «YES» (без кавычек), иначе выведите «NO» (без кавычек).

Формат входных данных

На вход программе подаётся натуральное число.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

2. Заданы две клетки шахматной доски. Напишите программу, которая определяет, имеют ли указанные клетки один цвет или нет. Если они покрашены в один цвет, то выведите слово «YES» (без кавычек), а если в разные цвета, то «NO» (без кавычек).

Формат входных данных

На вход программе подаются четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

3. Футбольная команда набирает девочек от 10 до 15 лет включительно. Напишите программу, которая запрашивает возраст и пол претендента, используя для обозначения пола буквы «m» (от male – мужчина) и «f» (от female – женщина), и определяет, подходит ли претен-

дент для вступления в команду или нет. Если претендент подходит, то выведите «YES» (без кавычек), иначе выведите «NO» (без кавычек).

Формат входных данных

На вход программе подаётся натуральное число – возраст претендента и буква обозначающая пол m (мужчина) или f (женщина).

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

4. Напишите программу, которая считывает целое число и выводит соответствующую ему римскую цифру. Если число находится вне диапазона [1;10], то программа должна вывести текст «ошибка» (без кавычек).

5. Напишите программу, которая принимает на вход число и в зависимости от условий выводит текст «YES» (без кавычек) либо «NO» (без кавычек).

Условия:

- если число нечётное, то вывести «YES»;
- если число чётное в диапазоне от 2 до 5 (включительно), то вывести «NO»;
- если число чётное в диапазоне от 6 до 20 (включительно), то вывести «YES»;
- если число чётное и больше 20, то вывести «NO».

Формат входных данных

На вход программе подаётся натуральное число.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

6. Даны две различные клетки шахматной доски. Напишите программу, которая определяет, может ли слон попасть с первой клетки на вторую одним ходом. Программа получает на вход четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки. Программа должна вывести «YES», если из первой клетки ходом слона можно попасть во вторую, или «NO» в противном случае.

Формат входных данных

На вход программе подаются четыре числа от 1 до 8.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

Примечание. Шахматный слон ходит по диагоналям.

7. Даны две различные клетки шахматной доски. Напишите программу, которая определяет, может ли конь попасть с первой клетки на вторую одним ходом. Программа получает на вход четыре числа от 1 до 8 каждое, задающие номер столбца и номер строки сначала для первой клетки, потом для второй клетки. Программа должна вывести «YES», если из первой клетки ходом коня можно попасть во вторую, или «NO» в противном случае.

Формат входных данных

На вход программе подаются четыре числа от 1 до 8.

Формат выходных данных

Программа должна вывести текст в соответствии с условием задачи.

3.2.3. Контрольные вопросы третьей аттестации

1. Какую последовательность чисел даст вам вызов функции range(6)?
2. Какую последовательность чисел даст вам вызов функции range(2, 6)?
3. Какую последовательность чисел даст вам вызов функции range(0, 501, 100)?
4. Какую последовательность чисел даст вам вызов функции range(10, 5, -1)?
5. Что покажет приведённый ниже код?

```
for i in range(10, 25):
```

```
    print("Python awesome!")
```

6. Определите, какую задачу решает приведённый ниже код?

```
n = int(input())
```



```

counter = 0
for i in range(1, n + 1):
    if i % 3 == 0 and i % 7 != 0:
        counter += 1
print(counter)

```

7. Какое число нужно написать вместо многоточия, чтобы цикл выполнялся ровно 7 раз?

```

i = ...
while i <= 10:
    print('Python!')
    i += 1

```

8. Определите, какую задачу решает приведенный ниже код?

```

n = int(input())
res = 1
i = 2
while i <= n:
    res *= i
    i += 1
print(res)

```

9. Определите, какую задачу решает приведенный ниже код?

```

n = int(input())
i = 2
while n % i != 0:
    i += 1
print(i)

```

3.3. Задания для промежуточной аттестации

3.3.1 Контрольные вопросы для проведения экзамена

1. Каковы основные особенности языка Python как интерпретируемого языка программирования? Перечислите не менее трех областей его применения.
2. В чем заключается философия Python (The Zen of Python)? Приведите три принципа и кратко их поясните.
3. Объясните разницу между операторами `//`, `%` и `**`. Приведите пример использования каждого из них.
4. Напишите фрагмент кода, который запрашивает у пользователя его имя и год рождения, а затем вычисляет и выводит его возраст.
5. Напишите код с использованием конструкции `if-elif-else`, который проверяет число, введенное пользователем, и выводит «Положительное», «Отрицательное» или «Ноль».
6. В чем ключевое различие между циклами `while` и `for`? Приведите пример задачи, для которой предпочтительнее использовать цикл `while`.
7. Объясните назначение и механизм работы операторов `break` и `continue` внутри цикла. Приведите простой пример для каждого.
8. Напишите код, который с помощью цикла `for` и условного оператора `if` выводит только четные числа из списка `[1, 2, 3, 4, 5, 6, 7, 8, 9, 10]`.
9. Опишите, как работает "бесконечный цикл" (`while True`) и каким оператором его можно безопасно прервать? Приведите пример считывания команд от пользователя до тех пор, пока он не введет "выход".
10. В чем разница между `pass`, `continue` и `break`? Приведите ситуацию, где использование `pass` является осознанным выбором.

11. Как с помощью вложенных циклов `for` можно заполнить матрицу (список списков) размером 3x3 так, чтобы элементы имели значения $i * j$ (где i - номер строки, j - номер столбца)? Напишите соответствующий код.

12. Чем принципиально отличаются списки (`list`) от кортежей (`tuple`)? В каком случае предпочтительнее использовать кортеж?

13. Дан список: `numbers = [1, 2, 3, 4, 5]`. Напишите код, который с помощью методов списка добавляет число 6 в конец, а затем удаляет элемент с значением 3.

14. Что такое множества (`set`) и для решения каких задач они наиболее эффективны? Продемонстрируйте операцию объединения двух множеств.

15. Как устроены словари (`dict`) в Python? Напишите пример словаря, содержащего информацию о студенте (имя, факультет, средний балл).

16. Дан кортеж `immutable = (1, 2, 3)`. Почему попытка выполнить операцию `immutable[0] = 10` приводит к ошибке? Какова практическая польза от этой "неизменяемости"?

17. Как с помощью метода словаря `get()` избежать возникновения ошибки `KeyError` при обращении к несуществующему ключу? Продемонстрируйте на примере: `data = {'a': 1}`, попытка получить значение по ключу 'b'.

18. Что выведет следующий код и почему? Объясните поведение операции со множествами.

```
python
set1 = {1, 2, 3, 4, 5}
set2 = {4, 5, 6, 7, 8}
print(set1 - set2)
print(set1 | set2)
```

19. Как с помощью спискового включения (`list comprehension`) создать список, содержащий квадраты чисел от 1 до 10? Приведите код.

20. Каким методом можно "распаковать" кортеж из трех элементов в три отдельные переменные? Напишите пример.

21. В чем разница между методами списков `append()`, `extend()` и `insert()`? Продемонстрируйте разницу на простом примере.

22. Опишите механизм передачи аргументов в функцию. В чем разница между позиционными и именованными аргументами?

23. Что такое `*args` и `**kwargs`? Для чего они используются при определении функции?

24. Как организовать импорт собственного модуля? Напишите пример создания простого модуля `geometry.py` с функцией `circle_area(r)` и ее импорта в основной файл.

25. Что такое область видимости (`scope`) переменной? Объясните на примере, что такое локальная и глобальная переменная.

26. Что такое рекурсия? Какая главная опасность связана с неправильной реализацией рекурсивной функции? Напишите простую рекурсивную функцию для вычисления факториала.

27. Можно ли изменить значение глобальной переменной внутри функции? Если да, то с помощью какого ключевого слова?

28. Что произойдет, если в функции, определенной как `def func(a, b=[])`, использовать изменяемый объект (например, список) в качестве значения по умолчанию? В чем заключается опасность?

29. Представьте, что вы создали модуль `calculator.py` с функциями `add(a, b)` и `multiply(a, b)`. Напишите три разных способа импорта этих функций в основной скрипт.

3.4.Задания для проверки остаточных знаний

3.4.1. Вопросы для проверки остаточных знаний

Раздел 1. Базовые конструкции языка Python

1. Синтаксис и типы данных

- о Какие основные типы данных в Python вы знаете? Приведите примеры каждого типа
- о В чем разница между списком (list) и кортежем (tuple)? Когда следует использовать каждый из них?

- о Что такое динамическая типизация? Как она реализована в Python?

2. Операторы и управляющие конструкции

- о Опишите различия между операторами == и is в Python
- о Как работает тернарный оператор? Приведите пример использования
- о В чем разница между циклами for и while? Когда следует использовать каждый из них?

Раздел 2. Функции и модульность

3. Функции

- о Что такое параметры по умолчанию? Приведите пример функции с параметрами по умолчанию
- о Как передаются аргументы в функции Python: по значению или по ссылке? Объясните на примере

- о Что такое lambda-функции? В каких случаях их целесообразно использовать?

4. Область видимости

- о Объясните разницу между локальными и глобальными переменными
- о Что такое пространство имен (namespace) в Python?
- о Как работает инструкция global? Приведите пример использования

Раздел 3. Структуры данных

5. Списки и словари

- о Какие методы для работы со списками вы помните? Опишите назначение 3-4 методов

- о Как создать словарь? Какие типы данных могут быть ключами словаря?

- о Что такое генераторы списков (list comprehensions)? Приведите пример

6. Множества и кортежи

- о В чем преимущества использования множеств (set) перед списками?
- о Можно ли изменить элемент кортежа после его создания? Обоснуйте ответ
- о Какие операции можно выполнять над множествами? Приведите примеры

Раздел 4. Работа с файлами и исключения

7. Файловые операции

- о Какие режимы открытия файлов вы знаете? В чем разница между 'r', 'w' и 'a'?
- о Что такое контекстный менеджер with? Какие преимущества его использования?
- о Как прочитать все строки файла в список?

8. Обработка исключений

- о Для чего используется конструкция try-except? Приведите пример
- о Что такое блок finally? Когда он выполняется?
- о Как создать пользовательское исключение?

Раздел 5. Объектно-ориентированное программирование

9. Классы и объекты

- о Что такое конструктор класса? Какой метод отвечает за создание объекта?
- о Объясните разницу между атрибутами класса и атрибутами экземпляра
- о Что такое наследование? Приведите пример иерархии классов

10. Инкапсуляция и полиморфизм

- о Как реализована инкапсуляция в Python? Что такое "protected" и "private" атрибуты?

- Что такое полиморфизм? Приведите пример полиморфного поведения методов
- Что такое магические методы? Назовите 3-4 наиболее часто используемых магических метода

Раздел 6. Практические задачи

11. Анализ кода

python

```
def process_data(data):
    result = []
    for item in data:
        if item % 2 == 0:
            result.append(item * 2)
        else:
            result.append(item + 1)
    return result
```

```
numbers = [1, 2, 3, 4, 5]
print(process_data(numbers))
```

- Что выведет эта программа? Объясните логику работы функции

12. Нахождение ошибок

python

```
class Calculator:
    def __init__(self, value=0):
        self.value = value

    def add(self, x):
        value += x

    def get_value(self):
        return self.value
```

```
calc = Calculator(5)
calc.add(3)
print(calc.get_value())
```

- Найдите и исправьте ошибки в коде. Объясните, в чем заключались ошибки
- ### **13. Написание кода**
- Напишите функцию, которая принимает список чисел и возвращает список квадратов четных чисел
 - Создайте класс "Студент" с атрибутами "имя", "возраст", "средний балл" и методом для вывода информации
 - Напишите программу, которая читает текстовый файл и подсчитывает количество слов в нем

Раздел 7. Модули и пакеты

14. Работа с модулями

- Как импортировать конкретную функцию из модуля?
- Что такое пакет в Python? Какова его структура?
- Для чего используется файл `__init__.py`?

15. Стандартные библиотеки

- Какие модули стандартной библиотеки Python для работы с датой и временем вы знаете?
- Какой модуль используется для работы с регулярными выражениями?
- Какие модули помогают в работе с операционной системой?

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

В качестве методического материала рекомендуется использовать:

1. Положение о ФОС в ФГБОУ ВО «Дагестанский государственный технический университет» (Приложение № 9 к ООП).
2. Положение ФГБОУ ВО «Дагестанский государственный технический университет» о модульно-рейтинговой системе оценки учебной деятельности студентов.
3. Процедура проведения оценочных мероприятий.

4.1. Процедура проведения оценочных мероприятий.

4.1.1. Текущий контроль представляет собой проверку усвоения учебного материала теоретического и практического характера, регулярно осуществляемую на протяжении семестра. К основным формам текущего контроля (текущей аттестации) можно отнести устный опрос, письменные задания, контрольные работы.

Основные этапы текущего контроля:

– в конце каждой лекции или практического занятия студентам выдаются задания для внеаудиторного выполнения по соответствующей теме;

– срок выполнения задания устанавливается по расписанию занятий (к очередной лекции или практическому занятию);

– студентам, пропускающим занятия, выдаются дополнительные задания – представить конспект пропущенного занятия, написанный «от руки» с последующим собеседованием по теме занятия;

– подведение итогов контроля проводится по графику проведения текущего контроля;

– результаты оценки успеваемости заносятся в рейтинговую ведомость и доводятся до сведения студентов;

– студентам не получившим зачетное количество баллов по текущему контролю выдается дополнительные задания на зачетном занятии в промежуточную аттестацию.

К достоинствам данного типа относится его систематичность, непосредственно коррелирующаяся с требованием постоянного и непрерывного мониторинга качества обучения, а также возможность балльно-рейтинговой оценки успеваемости обучающихся.

Недостатком является фрагментарность и локальность проверки. Компетенцию целиком, а не отдельные ее элементы (знания, умения, навыки) при подобном контроле проверить невозможно.

4.1.2. Промежуточная аттестация, как правило, осуществляется в конце семестра и может завершать изучение, как отдельной дисциплины, так и ее раздела (разделов).

Промежуточная аттестация помогает оценить более крупные совокупности знаний и умений, в некоторых случаях – даже формирование определенных профессиональных компетенций.

Достоинства: помогает оценить более крупные совокупности знаний и умений, в некоторых случаях – даже формирование определенных профессиональных компетенций.

Основные формы промежуточной аттестации: зачет и экзамен.

Текущий контроль и промежуточная аттестация традиционно служат основным средством обеспечения в учебном процессе «обратной связи» между преподавателем и обучающимся, необходимой для стимулирования работы обучающихся и совершенствования методики преподавания учебных дисциплин.

Основные этапы промежуточной аттестации:

- зачетное занятие (экзамен) проводится по расписанию сессии;
- форма проведения занятия – письменная контрольная работа;
- вид контроля – фронтальный;
- требование к содержанию контрольной работы – дать краткий ответ на поставленный вопрос (задание);
- количество вопросов в зачетном задании;
- итоговая оценка определяется как сумма оценок, полученных в текущей аттестации и по результатам написания контрольной работы;
- проверка ответов и объявление результатов производится в день написания контрольной работы;
- результаты аттестации заносятся в экзаменационно-зачетную ведомость и зачетную книжку студента (при получении зачета).

Студенты, не прошедшие промежуточную аттестацию по графику сессии, должны ликвидировать задолженность в установленном порядке.

При первой попытке ликвидации задолженности, во время зачетной недели или в течение сессии, студенту выдаются все задания по текущему контролю и промежуточной аттестации, по которым он не смог набрать зачетное количество баллов.

При ликвидации задолженности после сессии студенту выдаются для выполнения все задания по текущему контролю, кроме аналитического обзора, если он выполнен ранее, и вопросы зачетного занятия промежуточной аттестации, включая дополнительные вопросы по теме аналитического обзора.