

Документ подписан простой электронной подписью

Информация о владельце:

ФИО: Минцаев Магомед Шавалович

Должность: Ректор

Дата подписания: 05.06.2026 10:04:12

Уникальный программный ключ:

236bcc35c296f119d6aafdc22836b21db52dbc07971a86865a5825f9fa4304cc

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ**

**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА М.Д.МИЛЛИОНЩИКОВА»**

Информационные системы в экономике
(наименование кафедры)

(подпись)

УТВЕРЖДЕН
на заседании кафедры
«02» 09 2026 г., протокол № 1
Заведующий кафедрой
Л.Р. Магомаева

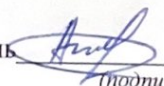
ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

«Разработка ПО на языке Python»
(наименование дисциплины)

Направление /специальность подготовки
09.03.03 Прикладная информатика
(код и наименование направления/ специальности подготовки)

Направленность (профиль)
Разработка мобильных и Web-приложений
(наименование специализации / профиля подготовки)

Квалификация
Бакалавр
(специалист / бакалавр / магистр)

Составитель  А. Ю. Ахматсултанов
(подпись)

ПАСПОРТ
ФОНДА ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ
«Разработка ПО на языке Python»

№ п/п	Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1.	Тема 1: Введение в программирование и Python	ПК – 5. ОПК-2.	Лабораторная работа 1. Основы программирования на Python
2.	Тема 2: Переменные и типы данных	ПК – 5.	Лабораторная работа 2. Работа с переменными и типами данных в Python
3.	Тема 3: Условные конструкции и Циклы	ПК – 5.	Лабораторная работа 3. Управление потоком программы в Python
4.	Тема 4: Списки и кортежи	ПК – 5. ОПК-2.	Лабораторная работа 4. Изучение списков и кортежей в Python
5.	Тема 5: Функции	ПК – 5.	Лабораторная работа 5. Основы работы с функциями в Python
6.	Тема 6: Работа с файлами	ПК – 5.	Лабораторная работа 6. Управление файлами в Python
7.	Тема 7: Итераторы и генераторы	ПК – 5. ОПК-2.	Лабораторная работа 7. Итераторы и генераторы в Python
8.	Тема 8: Регулярные выражения	ПК – 5.	Лабораторная работа 8. Освоение регулярных выражений в Python
9.	Тема 9: Введение в ООП	ПК – 5.	Лабораторная работа 9. Основы объектно-ориентированного программирования в Python

10.	Тема 10: Атрибуты, свойства и методы	ПК – 5. ОПК-2.	Лабораторная работа 10. Изучение атрибутов, свойств и методов в объектно-ориентированном программировании Python
11.	Тема 11: Магические методы	ПК – 5.	Лабораторная работа 11. Магические методы в Python
12.	Тема 12: Наследование, инкапсуляция и полиморфизм	ПК – 5.	Лабораторная работа 12. Наследование, инкапсуляция и полиморфизм в Python
13.	Тема 13: Модули и пакеты	ПК – 5. ОПК-2.	Лабораторная работа 13. Работа с модулями и пакетами в Python

ПЕРЕЧЕНЬ ОЦЕНОЧНЫХ СРЕДСТВ

№ п/п	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в фонде
1	<i>Лабораторная работа</i>	Средство проверки умений применять полученные знания по заранее определенной методике для решения задач или заданий по модулю или дисциплине в целом	Комплект заданий для выполнения лабораторных работ
2	<i>Рубежный контроль</i>	Форма проверки знаний по дисциплине в виде первой и второй рубежных аттестаций	Вопросы к аттестациям
3	<i>Зачет</i>	Итоговая форма оценки знаний	Вопросы к зачету

4	<i>Экзамен</i>	Итоговая форма оценки знаний	Вопросы к экзамену
---	----------------	------------------------------	--------------------

ЗАДАНИЯ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ

В качестве оценочных средств используются средства контроля выполнения лабораторных работ по дисциплине. Защита лабораторной работы – ответ на контрольные вопросы после выполнения лабораторной работы.

Средства текущего контроля: устный опрос (собеседование/опрос, разбор учебной ситуации на выбранную тему, подготовка устных сообщений и докладов), практическое задание (выполнение заданий в электронной форме на ПК).

Лабораторная работа 1. Основы программирования на Python	ПК – 5. ОПК-2.
Лабораторная работа 2. Работа с переменными и типами данных в Python	ПК – 5.
Лабораторная работа 3. Управление потоком программы в Python	ПК – 5.
Лабораторная работа 4. Изучение списков и кортежей в Python	ПК – 5. ОПК-2.
Лабораторная работа 5. Основы работы с функциями в Python	ПК – 5.
Лабораторная работа 6. Управление файлами в Python	ПК – 5.
Лабораторная работа 7. Итераторы и генераторы в Python	ПК – 5. ОПК-2.
Лабораторная работа 8. Освоение регулярных выражений в Python	ПК – 5.
Лабораторная работа 9. Основы объектно-ориентированного программирования в Python	ПК – 5.
Лабораторная работа 10. Изучение атрибутов, свойств и методов в объектно-ориентированном программировании Python	ПК – 5. ОПК-2.
Лабораторная работа 11. Магические методы в Python	ПК – 5.

Лабораторная работа 12. Наследование, инкапсуляция и полиморфизм в Python

ПК – 5.

Лабораторная работа 13. Работа с модулями и пакетами в Python

ПК – 5. ОПК-2.

Критерии оценки ответов на лабораторные работы (Зачет)

Регламентом БРС предусмотрено всего 15 баллов за текущую работу студента. Критерии оценки разработаны, исходя из возможности ответа студентом до 8 лабораторных работ с использованием дополнительного материала по ним. (до 4 балла).

4 балла ставится за работу, выполненную полностью без ошибок и недочетов.

3 балл ставится за работу, выполненную полностью, но при наличии в ней недочетов.

2 балл ставится за работу, выполненную полностью, но при наличии в ней негрубых ошибок и недочетов.

1 балл ставится за работу, выполненную полностью, но при наличии в ней грубых ошибок и недочетов.

0 баллов ставится, если студент совсем не выполнил ни одного задания.

Максимальное количество баллов (за весь семестр) за выполнение и защиту лабораторных работ - 30.

Максимально высокие баллы за текущий контроль – 15.

Критерии оценки ответов на лабораторные работы (Экзамен)

Регламентом БРС предусмотрено всего 15 баллов за текущую работу студента. Критерии оценки разработаны, исходя из возможности ответа студентом до 5 лабораторных работ с использованием дополнительного материала по ним. (до 8 балла).

8 балла ставится за работу, выполненную полностью без ошибок и недочетов.

6 балл ставится за работу, выполненную полностью, но при наличии в ней недочетов.

4 балл ставится за работу, выполненную полностью, но при наличии в ней негрубых ошибок и недочетов.

2 балл ставится за работу, выполненную полностью, но при наличии в ней грубых ошибок и недочетов.

0 баллов ставится, если студент совсем не выполнил ни одного задания.

Максимальное количество баллов (за весь семестр) за выполнение и защиту лабораторных работ - 30.

Максимально высокие баллы за текущий контроль – 15.

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА М.Д.МИЛЛИОНЩИКОВА**

**Институт цифровой экономики и технологического предпринимательства
Кафедра «Информационные системы в экономике»**

Вопросы к первой рубежной аттестации «Разработка ПО на языке Python» (Зачет/4-й семестр)

1. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.
2. Опишите, как развивался Python с момента его создания до наших дней, и укажите основные этапы его эволюции.
3. Объясните, как в Python используются переменные и какие правила следует соблюдать при их объявлении.
4. Сравните типы данных в Python (целые числа, вещественные числа, строки, булевы значения) с типами данных в других языках программирования.
5. Обсудите процесс и необходимость преобразования типов в Python, приведя примеры кода, где это преобразование критично.
6. Опишите, какие операции можно проводить над различными типами данных и как они влияют на результаты программы.
7. Подробно опишите, как работают условные конструкции в Python и в чем их важность при создании алгоритмов.
8. Разъясните использование циклов for и while в Python на примерах кода, объясняя, в каких случаях предпочтительнее использовать каждый из них.
9. Обсудите роль операторов break, continue и else в управлении циклами в Python, приведя примеры, когда их применение оправдано.
10. Опишите процесс создания и использования списков и кортежей в Python, указав ключевые различия и сценарии их применения.
11. Рассмотрите методы и функции для работы со списками в Python, объяснив, как каждый метод изменяет данные списка.
12. Объясните, как осуществляется индексация и работа со срезами в списках Python, и как эти возможности можно использовать для обработки данных.

Вопросы ко второй рубежной аттестации «Разработка ПО на языке Python» (Зачет/4-й семестр)

1. Каковы основы определения и вызова функций в Python? Проиллюстрируйте свой ответ примерами, подчеркивая важность правильного использования функций в программировании.
2. Объясните, как в Python работают параметры и аргументы функций, включая различия между позиционными и ключевыми аргументами.
3. Расскажите о различных типах возвращаемых значений в функции, и как возвращаемые значения влияют на поток данных в программе.
4. Проанализируйте концепцию областей видимости переменных в Python, объясняя, как локальная и глобальная области видимости влияют на данные.
5. Изложите, как и почему используются анонимные функции (лямбды) в Python, приведя примеры их применения для решения программных задач.
6. Опишите процесс открытия и закрытия файлов в Python, подчеркивая важность правильного управления файловыми ресурсами.
7. Объясните методы чтения и записи данных в файлы, включая примеры кода, которые иллюстрируют эти процессы.
8. Рассмотрите использование менеджера контекста `with` при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.
9. Объясните, что такое итераторы в Python, и как они используются для управления потоками данных.
10. Расскажите, как создаются и используются генераторы в Python, включая примеры функций и выражений, которые позволяют генерировать последовательности данных.
11. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.
12. Опишите основы синтаксиса регулярных выражений и как они применяются для обработки текстов в Python.
13. Проанализируйте функции модуля `re`, такие как поиск, замена и разделение строк, с примерами их использования.
14. Обсудите реальные сценарии использования регулярных выражений в программировании, демонстрируя, как они могут упростить и ускорить обработку данных.

Вопросы к первой рубежной аттестации «Разработка ПО на языке Python» (Экзамен/5-й семестр)

1. Опишите основные концепции объектно-ориентированного программирования и их значение в разработке современного программного обеспечения.
2. Как классы и объекты реализуются в Python? Приведите примеры создания классов и их использования через объекты.
3. Объясните, как работают конструкторы и деструкторы в классах Python, и какую роль они играют в управлении жизненным циклом объектов.
4. Расскажите о важности и применении атрибутов классов и экземпляров в Python, давая примеры из практической разработки.
5. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора `property` для создания управляемых свойств.
6. Опишите различие и применение статических методов и методов класса. В чем их уникальность и когда они могут быть особенно полезны?
7. Как механизм наследования в Python помогает в повторном использовании кода? Приведите примеры с множественным наследованием.
8. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.
9. Как реализуется полиморфизм в Python? Дайте примеры, показывающие, как один интерфейс может использоваться для различных типов данных.
10. В чем заключается польза и методика создания пакетов в Python? Объясните процесс структурирования больших проектов с помощью пакетов и модулей.
11. Какие проблемы могут возникнуть при работе с наследованием и как Python позволяет их решать? Обсудите конкретные случаи, связанные с разрешением методов в иерархиях классов.
12. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?

Вопросы ко второй рубежной аттестации «Разработка ПО на языке Python»
(Экзамен/5-й семестр)

1. Опишите, как магические методы `__init__`, `__str__`, и `__repr__` используются в Python для управления поведением объектов при создании, преобразовании в строку и официальном представлении объектов. Приведите примеры, иллюстрирующие их использование.
2. Расскажите о роли магических методов для операций сравнения в Python, таких как `__eq__`, `__lt__`, и `__gt__`. Как эти методы влияют на функциональность классов?
3. Объясните, как магические методы используются для работы с контейнерами в Python. Приведите примеры методов `__len__` и `__getitem__` и объясните, как они влияют на поведение классов, реализующих контейнеры.
4. Обсудите принципы наследования в объектно-ориентированном программировании с примерами из Python. Как наследование способствует повторному использованию кода?
5. Разъясните, как реализовано множественное наследование в Python и какие вызовы могут возникнуть при его использовании. Приведите примеры кода.
6. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?
7. Объясните, как реализуется полиморфизм в Python. Приведите примеры, показывающие, как один интерфейс может использоваться для различных типов данных.
8. Опишите процесс импорта стандартных и пользовательских модулей в Python. Какие проблемы могут возникнуть при неправильном управлении импортами?
9. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?
10. Обсудите, как пакеты и пространства имен помогают управлять большими программными проектами в Python. Приведите примеры, как структурирование кода в пакеты улучшает модульность и предотвращает конфликты имен.
11. Как магические методы влияют на поведение объектов при взаимодействии с Python системами, такими как циклы, функции и даже с самим интерпретатором Python? Приведите примеры использования магических методов в реальных приложениях.
12. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линейизации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?

13. Обсудите преимущества и недостатки использования полиморфизма в объектно-ориентированном программировании. Как полиморфизм может улучшить или усложнить код в больших программных проектах?

14. Какие практические меры можно предпринять для оптимизации структуры и производительности Python-программ, используя модули и пакеты? Приведите примеры, как организация кода в модули может помочь в оптимизации и упрощении сопровождения кода.

Критерии оценки ответов на рубежной аттестации

Регламентом БРС предусмотрено всего 20 баллов за рубежную аттестацию студента. Критерии оценки разработаны, исходя из возможности ответа студентом на 2 вопроса в билете (по 10 баллов).

10 баллов (5+) заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную и дополнительную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, разбирающийся в основных научных концепциях по изучаемой дисциплине, проявивший творческие способности и научный подход в понимании и изложении учебного программного материала, ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично.

9 баллов (5) заслуживает студент, обнаруживший всестороннее, систематическое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную литературу и знаком с дополнительной литературой, рекомендованной программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению, ответ отличается точностью использованных терминов, материал излагается последовательно и логично.

8 баллов (4+) заслуживает студент, обнаруживший полное знание учебно-программного материала, не допускающий в ответе существенных неточностей,

самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

7 баллов (4) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

6 баллов (4-) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, отличавшийся достаточной активностью на практических (семинарских) и лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы.

5 баллов (3+) заслуживает студент, обнаруживший знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для их самостоятельного устранения.

4 балла (3) заслуживает студент, обнаруживший знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их

выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя допущенных погрешностей.

3 балла (3-) заслуживает студент, обнаруживший знание основного учебно-программного материала в объёме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, однако допустивший погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей.

2 балла (2) выставляется студенту, обнаружившему пробелы в знаниях или отсутствие знаний по значительной части основного учебно-программного материала, не выполнившему самостоятельно предусмотренные программой основные задания, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, не отработавшему основные практические, семинарские, лабораторные занятия, допускающему существенные ошибки при ответе, и который не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по соответствующей дисциплине.

1 балл — нет ответа (отказ от ответа, представленный ответ полностью не по существу содержащихся в экзаменационном задании вопросов)

**ГРОЗНЕНСКИЙ ГОСУДАРСТВЕННЫЙ НЕФТЯНОЙ ТЕХНИЧЕСКИЙ
УНИВЕРСИТЕТ ИМЕНИ АКАДЕМИКА М.Д.МИЛЛИОНЩИКОВА**

**Институт цифровой экономики и технологического предпринимательства
Кафедра информационных систем в экономике**

Вопросы к зачету по дисциплине «Разработка ПО на языке Python»

1. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.
2. Опишите, как развивался Python с момента его создания до наших дней, и укажите основные этапы его эволюции.
3. Объясните, как в Python используются переменные и какие правила следует соблюдать при их объявлении.
4. Сравните типы данных в Python (целые числа, вещественные числа, строки, булевы значения) с типами данных в других языках программирования.
5. Обсудите процесс и необходимость преобразования типов в Python, приведя примеры кода, где это преобразование критично.
6. Опишите, какие операции можно проводить над различными типами данных и как они влияют на результаты программы.
7. Подробно опишите, как работают условные конструкции в Python и в чем их важность при создании алгоритмов.
8. Разъясните использование циклов for и while в Python на примерах кода, объясняя, в каких случаях предпочтительнее использовать каждый из них.
9. Обсудите роль операторов break, continue и else в управлении циклами в Python, приведя примеры, когда их применение оправдано.
10. Опишите процесс создания и использования списков и кортежей в Python, указав ключевые различия и сценарии их применения.
11. Рассмотрите методы и функции для работы со списками в Python, объяснив, как каждый метод изменяет данные списка.
12. Объясните, как осуществляется индексация и работа со срезами в списках Python, и как эти возможности можно использовать для обработки данных.

13. Каковы основы определения и вызова функций в Python? Проиллюстрируйте свой ответ примерами, подчеркивая важность правильного использования функций в программировании.
14. Объясните, как в Python работают параметры и аргументы функций, включая различия между позиционными и ключевыми аргументами.
15. Расскажите о различных типах возвращаемых значений в функциях, и как возвращаемые значения влияют на поток данных в программе.
16. Проанализируйте концепцию областей видимости переменных в Python, объясняя, как локальная и глобальная области видимости влияют на данные.
17. Изложите, как и почему используются анонимные функции (лямбды) в Python, приведя примеры их применения для решения программных задач.
18. Опишите процесс открытия и закрытия файлов в Python, подчеркивая важность правильного управления файловыми ресурсами.
19. Объясните методы чтения и записи данных в файлы, включая примеры кода, которые иллюстрируют эти процессы.
20. Рассмотрите использование менеджера контекста `with` при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.
21. Объясните, что такое итераторы в Python, и как они используются для управления потоками данных.
22. Расскажите, как создаются и используются генераторы в Python, включая примеры функций и выражений, которые позволяют генерировать последовательности данных.
23. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.
24. Опишите основы синтаксиса регулярных выражений и как они применяются для обработки текстов в Python.
25. Проанализируйте функции модуля `re`, такие как поиск, замена и разделение строк, с примерами их использования.
26. Обсудите реальные сценарии использования регулярных выражений в программировании, демонстрируя, как они могут упростить и ускорить обработку данных.

Критерии оценки ответов на зачете

Регламентом БРС предусмотрено всего 20 баллов за экзамен. Критерии оценки разработаны, исходя из возможности ответа студентом на 2 вопроса в билете (по 10 баллов).

10 баллов (5+) заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную и дополнительную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, разбирающийся в основных научных концепциях по изучаемой дисциплине, проявивший творческие способности и научный подход в понимании и изложении учебного программного материала, ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично.

9 баллов (5) заслуживает студент, обнаруживший всестороннее, систематическое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную литературу и знаком с дополнительной литературой, рекомендованной программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению, ответ отличается точностью использованных терминов, материал излагается последовательно и логично.

8 баллов (4+) заслуживает студент, обнаруживший полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

7 баллов (4) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно

работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

6 баллов (4-) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, отличавшийся достаточной активностью на практических (семинарских) и лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы.

5 баллов (3+) заслуживает студент, обнаруживший знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для их самостоятельного устранения.

4 балла (3) заслуживает студент, обнаруживший знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя допущенных погрешностей.

3 балла (3-) заслуживает студент, обнаруживший знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, однако допустивший погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей.

2 балла (2) выставляется студенту, обнаружившему пробелы в знаниях или отсутствие знаний по значительной части основного учебно-программного материала, не

выполнившему самостоятельно предусмотренные программой основные задания, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, не отработавшему основные практические, семинарские, лабораторные занятия, допускающему существенные ошибки при ответе, и который не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по соответствующей дисциплине.

1 балл — нет ответа (отказ от ответа, представленный ответ полностью не по существу содержащихся в экзаменационном задании вопросов)

Критерии оценки знаний на зачете

Оценку **"Зачтено"** заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка **"отлично"** выставляется студентам, усвоившим взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала.

Оценка **"Не зачтено"** выставляется студенту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий. Как правило, оценка **"неудовлетворительно"** ставится студентам, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.

Вопросы к экзамену по дисциплине «Разработка ПО на языке Python»

1. Опишите основные концепции объектно-ориентированного программирования и их значение в разработке современного программного обеспечения.
2. Как классы и объекты реализуются в Python? Приведите примеры создания классов и их использования через объекты.
3. Объясните, как работают конструкторы и деструкторы в классах Python, и какую роль они играют в управлении жизненным циклом объектов.

4. Расскажите о важности и применении атрибутов классов и экземпляров в Python, давая примеры из практической разработки.
5. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора `property` для создания управляемых свойств.
6. Опишите различие и применение статических методов и методов класса. В чем их уникальность и когда они могут быть особенно полезны?
7. Как механизм наследования в Python помогает в повторном использовании кода? Приведите примеры с множественным наследованием.
8. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.
9. Как реализуется полиморфизм в Python? Дайте примеры, показывающие, как один интерфейс может использоваться для различных типов данных.
10. В чем заключается польза и методика создания пакетов в Python? Объясните процесс структурирования больших проектов с помощью пакетов и модулей.
11. Какие проблемы могут возникнуть при работе с наследованием и как Python позволяет их решать? Обсудите конкретные случаи, связанные с разрешением методов в иерархиях классов.
12. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?
13. Опишите, как магические методы `__init__`, `__str__`, и `__repr__` используются в Python для управления поведением объектов при создании, преобразовании в строку и официальном представлении объектов. Приведите примеры, иллюстрирующие их использование.
14. Расскажите о роли магических методов для операций сравнения в Python, таких как `__eq__`, `__lt__`, и `__gt__`. Как эти методы влияют на функциональность классов?
15. Объясните, как магические методы используются для работы с контейнерами в Python. Приведите примеры методов `__len__` и `__getitem__` и объясните, как они влияют на поведение классов, реализующих контейнеры.

16. Обсудите принципы наследования в объектно-ориентированном программировании с примерами из Python. Как наследование способствует повторному использованию кода?
17. Разъясните, как реализовано множественное наследование в Python и какие вызовы могут возникнуть при его использовании. Приведите примеры кода.
18. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?
19. Объясните, как реализуется полиморфизм в Python. Приведите примеры, показывающие, как один интерфейс может использоваться для различных типов данных.
20. Опишите процесс импорта стандартных и пользовательских модулей в Python. Какие проблемы могут возникнуть при неправильном управлении импортами?
21. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?
22. Обсудите, как пакеты и пространства имен помогают управлять большими программными проектами в Python. Приведите примеры, как структурирование кода в пакеты улучшает модульность и предотвращает конфликты имен.
23. Как магические методы влияют на поведение объектов при взаимодействии с Python системами, такими как циклы, функции и даже с самим интерпретатором Python? Приведите примеры использования магических методов в реальных приложениях.
24. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линеаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?
25. Обсудите преимущества и недостатки использования полиморфизма в объектно-ориентированном программировании. Как полиморфизм может улучшить или усложнить код в больших программных проектах?
26. Какие практические меры можно предпринять для оптимизации структуры и производительности Python-программ, используя модули и пакеты? Приведите примеры, как организация кода в модули может помочь в оптимизации и упрощении сопровождения кода.

Критерии оценки ответов на экзамене

Регламентом БРС предусмотрено всего 20 баллов за экзамен. Критерии оценки разработаны, исходя из возможности ответа студентом на 2 вопроса в билете (по 10 баллов).

10 баллов (5+) заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную и дополнительную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, разбирающийся в основных научных концепциях по изучаемой дисциплине, проявивший творческие способности и научный подход в понимании и изложении учебного программного материала, ответ отличается богатством и точностью использованных терминов, материал излагается последовательно и логично.

9 баллов (5) заслуживает студент, обнаруживший всестороннее, систематическое знание учебного программного материала, самостоятельно выполнивший все предусмотренные программой задания, глубоко усвоивший основную литературу и знаком с дополнительной литературой, рекомендованной программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению, ответ отличается точностью использованных терминов, материал излагается последовательно и логично.

8 баллов (4+) заслуживает студент, обнаруживший полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

7 баллов (4) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший все предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, активно работавший на практических, семинарских, лабораторных занятиях, показавший систематический характер знаний по

дисциплине, достаточный для дальнейшей учебы, а также способность к их самостоятельному пополнению.

6 баллов (4-) заслуживает студент, обнаруживший достаточно полное знание учебно-программного материала, не допускающий в ответе существенных неточностей, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, отличавшийся достаточной активностью на практических (семинарских) и лабораторных занятиях, показавший систематический характер знаний по дисциплине, достаточный для дальнейшей учебы.

5 баллов (3+) заслуживает студент, обнаруживший знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для их самостоятельного устранения.

4 балла (3) заслуживает студент, обнаруживший знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, усвоивший основную литературу, рекомендованную программой, однако допустивший некоторые погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя допущенных погрешностей.

3 балла (3-) заслуживает студент, обнаруживший знание основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, не отличавшийся активностью на практических (семинарских) и лабораторных занятиях, самостоятельно выполнивший основные предусмотренные программой задания, однако допустивший погрешности при их выполнении и в ответе на экзамене, но обладающий необходимыми знаниями для устранения под руководством преподавателя наиболее существенных погрешностей.

2 балла (2) выставляется студенту, обнаружившему пробелы в знаниях или отсутствие знаний по значительной части основного учебно-программного материала, не выполнившему самостоятельно предусмотренные программой основные задания, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, не отработавшему основные практические, семинарские, лабораторные занятия, допускающему существенные ошибки при ответе, и который не может продолжить обучение или приступить к профессиональной деятельности без дополнительных занятий по соответствующей дисциплине.

1 балл — нет ответа (отказ от ответа, представленный ответ полностью не по существу содержащихся в экзаменационном задании вопросов)

Критерии оценки знаний на экзамене

Оценку **"отлично"** заслуживает студент, обнаруживший всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, усвоивший основную и знакомый с дополнительной литературой, рекомендованной программой. Как правило, оценка "отлично" выставляется студентам, усвоившим взаимосвязь основных понятий дисциплины в их значении для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала.

Оценку **"хорошо"** заслуживает студент, обнаруживший полное знание учебно-программного материала, успешно выполняющий предусмотренные в программе задания, усвоивший основную литературу, рекомендованную в программе. Как правило, оценка "хорошо" выставляется студентам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному пополнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности.

Оценку **"удовлетворительно"** заслуживает студент, обнаруживший знания основного учебно-программного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по специальности, справляющийся с выполнением заданий, предусмотренных программой, знакомый с основной литературой, рекомендованной программой. Как правило, оценка "удовлетворительно" выставляется студентам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя.

Оценка **"неудовлетворительно"** выставляется студенту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий. Как правило, оценка "неудовлетворительно" ставится студентам, которые не могут продолжить обучение или приступить к профессиональной деятельности по окончании вуза без дополнительных занятий по соответствующей дисциплине.

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ВЫПОЛНЕНИЯ ЛАБОРАТОРНЫХ РАБОТ

Лабораторная работа 1. Основы программирования на Python

Название: Основы программирования на Python

Цель: познакомить студентов с основами программирования и языком Python, научить устанавливать Python и запускать базовые программы.

Задания:

1. **Установка Python**

- Установите Python последней версии с официального сайта.
- Установите текстовый редактор или интегрированную среду разработки (IDE), например, Visual Studio Code или PyCharm.

2. **Настройка рабочей среды**

- Настройте IDE для работы с Python.
- Проверьте корректность установки, запустив интерпретатор Python из терминала или консоли.

3. **Запуск первой программы**

- Создайте новый файл с расширением .py.
- Напишите программу, которая выводит на экран фразу "Привет, мир!".
- Запустите программу и убедитесь, что она корректно выполняет вывод в консоль.

Контрольные вопросы:

1. Что такое программирование и каковы его основные цели?
2. Какие особенности Python выделяют его среди других языков программирования?
3. Какие шаги нужно выполнить для установки Python на вашу операционную систему?
4. Какова роль текстового редактора или IDE в процессе программирования?
5. Какие ошибки могут возникнуть при запуске первой программы и как их исправить?

Лабораторная работа 2. Работа с переменными и типами данных в Python

Название: Работа с переменными и типами данных в Python

Цель: Научить студентов создавать переменные, использовать различные типы данных и осуществлять базовые операции над данными в Python.

Задания:

1. **Создание переменных**

- Создайте переменные для хранения вашего имени, возраста, роста и студенческого статуса (булево значение).

- Выведите значения всех переменных на экран.

2. Операции с числами

- Создайте переменные для хранения двух чисел.

- Выполните и выведите результаты основных арифметических операций (сложение, вычитание, умножение, деление, возведение в степень).

3. Работа со строками

- Создайте строковые переменные и соедините их в одну строку (конкатенация).

- Примените к результату методы `.upper()`, `.lower()`, и `.title()`, выводя результаты на экран.

4. Преобразование типов

- Преобразуйте строку, содержащую число, в целое число и в вещественное число.

- Попробуйте преобразовать логическое значение в строку и обратно.

- **Исследование операций**

- Используйте сложные математические функции, например, `sqrt` или `log` из модуля `math` для работы с числами.

- Попробуйте форматировать вывод чисел с использованием строкового метода `.format()` или f-строк.

Контрольные вопросы:

1. Что такое переменная в контексте программирования на Python?

2. Какие типы данных вам известны и какие операции можно с ними выполнять?

3. Как произвести преобразование типов и для чего это может потребоваться?

4. Какие ошибки можно встретить при работе с разными типами данных?

5. Какие математические функции Python вы использовали и в каких случаях они применимы?

Лабораторная работа 3. Управление потоком программы в Python

Название: Управление потоком программы в Python

Цель: Научить студентов использовать условные конструкции и циклы для контроля за выполнением программ, а также освоить итерацию по различным последовательностям.

Задания:

1. **Использование условных конструкций**

- Напишите программу, которая запрашивает у пользователя его возраст и сообщает, имеет ли он право голоса (возраст 18+).

2. **Итерация с использованием цикла for**

- Создайте список с несколькими элементами разных типов.
- Напишите цикл for, который выводит каждый элемент списка, а также его тип.

3. **Использование цикла while**

- Напишите программу, которая начинается с числа 1 и умножает его на 2 до тех пор, пока результат не станет больше 1000, выводя каждый результат на экран.

4. **Управление выполнением циклов**

- Используйте break и continue для управления циклом for, который итерирует числа от 1 до 20 и пропускает все числа, кратные 4, но прерывает цикл, если число равно 15.

5. **Продвинутое использование циклов**

- Напишите цикл, который ищет первый элемент в списке, который соответствует определённому условию (например, первое чётное число) и использует else для вывода сообщения, если такой элемент не найден.

Контрольные вопросы:

1. Как работают условные операторы в Python и для чего они используются?
2. В чём разница между циклами for и while?
3. Какие сценарии наиболее подходят для использования ключевых слов break, continue и else в циклах?
4. Приведите пример программы, где использование цикла while более целесообразно, чем for.
5. Как можно использовать условные конструкции и циклы для обработки данных, полученных от пользователя?

Лабораторная работа 4. Изучение списков и кортежей в Python

Название: Изучение списков и кортежей в Python

Цель: Познакомить студентов с основами работы со списками и кортежами, их методами и функциями, а также научить их использовать индексацию и срезы для доступа и манипуляции данными.

Задания:

1. **Создание списков и кортежей**

- Создайте список, содержащий различные типы данных (числа, строки, булевы значения).

- Создайте кортеж, содержащий несколько элементов (например, разные виды фруктов).

- Выведите оба созданных объекта.

2. Работа с методами списков

- Используйте методы `.append()`, `.remove()`, `.sort()`, и `.reverse()` для изменения и организации элементов списка.

- Создайте новый список и продемонстрируйте использование метода `.copy()` для копирования данных.

3. Индексация и срезы

- Получите доступ к отдельным элементам в списке и кортеже через индексацию.

- Создайте срезы списка: выберите первые три элемента, последний элемент, а также все элементы с шагом два.

4. Преобразование списков в кортежи и обратно

- Преобразуйте созданный ранее список в кортеж, а кортеж — в список.

- Проверьте, изменились ли объекты после преобразования.

5. Продвинутое манипулирование списками

- Используйте списковые включения (list comprehensions) для создания нового списка, где каждый элемент является результатом операции над элементами исходного списка.

Контрольные вопросы:

1. В чем различие между списками и кортежами в Python?
2. Какие методы списков вы знаете и как они функционируют?
3. Как можно использовать индексацию и срезы для работы с последовательностями?
4. Какие операции можно выполнить при помощи списковых включений?
5. Почему и когда стоит использовать кортежи вместо списков?

Лабораторная работа 5. Основы работы с функциями в Python

Название: Основы работы с функциями в Python

Цель: Научить студентов создавать и использовать функции в Python, включая понимание параметров, аргументов, возвращаемых значений, областей видимости переменных и анонимных функций.

Задания:

1. Создание и вызов функций

- Напишите функцию, которая принимает имя пользователя и выводит приветственное сообщение.

- Вызовите эту функцию с несколькими разными именами.

2. **Параметры и аргументы функций**

- Создайте функцию, которая принимает несколько параметров (например, имя и возраст), и выводит эту информацию в форматированном виде.

- Вызовите функцию, передавая аргументы как по позиции, так и по ключу.

3. **Возвращаемые значения**

- Разработайте функцию для вычисления факториала числа и возврата результата.

- Протестируйте функцию с несколькими значениями и выведите результаты.

4. **Области видимости переменных**

- Определите переменные в различных областях видимости: глобальной, локальной и внутри блока функции.

- Продемонстрируйте, как изменения в этих переменных влияют на другие части кода.

5. **Анонимные функции (лямбды)**

- Создайте анонимную функцию для выполнения простой арифметической операции (например, умножения).

- Используйте эту функцию в сочетании с функциями высшего порядка, такими как `map()` или `filter()`, для обработки коллекций данных.

Контрольные вопросы:

1. Что такое функция в Python и какие задачи она помогает решать?
2. Какие типы аргументов могут быть у функции и как передавать их в функцию?
3. Как функция в Python возвращает значение и как это значение можно использовать?
4. Как работает область видимости переменных в контексте функций?
5. Что такое анонимные функции и каковы примеры их использования?

Лабораторная работа 6. Управление файлами в Python

Название: Управление файлами в Python

Цель: Научить студентов основам работы с файлами в Python, включая открытие, чтение, запись и использование менеджера контекста для эффективного управления ресурсами.

Задания:

1. **Открытие и закрытие файлов**

- Напишите программу, которая открывает файл в режиме чтения, считывает содержимое и закрывает файл. Убедитесь, что файл закрывается даже в случае возникновения ошибок.

2. **Чтение данных из файла**

- Создайте текстовый файл с несколькими строками текста.

- Реализуйте функцию, которая читает файл построчно и выводит каждую строку с её номером на экран.

3. **Запись данных в файл**

- Напишите функцию, которая принимает текст от пользователя и записывает этот текст в файл. После записи проверьте содержимое файла.

4. **Работа с файлами через менеджер контекста with**

- Используйте конструкцию with для безопасного открытия и автоматического закрытия файла после завершения работы.

- Продемонстрируйте чтение и запись файла с помощью with.

5. **Расширенное задание: работа с двоичными файлами**

- Откройте изображение в двоичном режиме и прочитайте первые 100 байт.

- Запишите эти байты в новый файл и проверьте, что создан корректный копированный файл.

Контрольные вопросы:

1. Какие режимы открытия файлов доступны в Python и для чего каждый из них используется?

2. Что происходит, если попытаться читать из файла, открытого только для записи, или писать в файл, открытый только для чтения?

3. Каковы преимущества использования менеджера контекста with при работе с файлами?

4. Как можно обработать ошибки при работе с файлами, чтобы предотвратить потерю данных или повреждение файлов?

5. В чем разница между работой с текстовыми и двоичными файлами?

Лабораторная работа 7. Итераторы и генераторы в Python

Название: Итераторы и генераторы в Python

Цель: Обучить студентов основам работы с итераторами и генераторами в Python, показать их преимущества и научить создавать генераторы с помощью функций и выражений.

Задания:

1. **Использование итераторов**

- Создайте список или кортеж с различными элементами.

- Получите итератор с помощью функции iter() и проитерируйте его элементы с помощью функции next() до возникновения исключения StopIteration.

2. **Создание генератора с помощью функции**

- Напишите функцию-генератор, которая генерирует квадраты чисел от 1 до заданного значения.

- Проитерируйте полученный генератор и выведите результаты на экран.

3. Создание генератора с помощью генераторного выражения

- Используйте генераторное выражение для создания генератора, который генерирует четные числа в заданном диапазоне.

- Проитерируйте этот генератор и выведите его элементы.

4. Преимущества использования генераторов

- Сравните использование генератора с аналогичной функцией, которая возвращает список. Оцените разницу в использовании памяти и времени выполнения.

5. Продвинутое задание: Генератор для чтения файлов

- Напишите генератор, который читает большой текстовый файл построчно и возвращает строки, содержащие определенное слово.

- Протестируйте его на примере большого файла, проверив его эффективность и удобство использования.

Контрольные вопросы:

1. Что такое итератор и как он работает в Python?

2. Как создать и использовать генератор в Python?

3. В чем заключаются преимущества использования генераторов по сравнению с другими структурами данных?

4. Какие задачи лучше всего решать с помощью генераторов?

5. Какие ресурсы и методы можно использовать для эффективной работы с большими объемами данных при помощи генераторов?

Лабораторная работа 8. Освоение регулярных выражений в Python

Название: Освоение регулярных выражений в Python

Цель: Познакомить студентов с синтаксисом и функциями модуля `re` для работы с регулярными выражениями в Python, научить их использовать эти инструменты для поиска, замены и разделения текстовых данных.

Задания:

1. Основы синтаксиса регулярных выражений

- Напишите регулярные выражения для проверки корректности формата email адресов, дат (например, формат "дд.мм.гггг") и номеров телефонов.

2. Функция поиска

- Используйте функцию `re.search()` для нахождения первого вхождения шаблона в тексте. Например, найдите первое упоминание слова "Python" в статье о программировании.

3. **Функция замены**

- Примените функцию `re.sub()` для замены всех вхождений определенного шаблона в строке. Например, замените все упоминания "Python" на "Python (популярный язык программирования)".

4. **Функция разделения**

- Используйте `re.split()` для разделения текста на предложения или другие компоненты на основе регулярных выражений, например, разделяя текст по запятым или точкам.

5. **Продвинутое использование регулярных выражений**

- Составьте регулярное выражение для извлечения всех URL из заданного текста.
- Используйте регулярные выражения для валидации сложных условий, например, проверки правильности пароля, который должен содержать буквы, цифры и специальные символы.

Контрольные вопросы:

1. Что такое регулярные выражения и каковы основы их синтаксиса?
2. Какие методы предоставляет модуль `re` и для чего они используются?
3. В каких ситуациях регулярные выражения являются наиболее полезными?
4. Какие могут быть сложности при работе с регулярными выражениями и как их избежать?
5. Как можно оптимизировать производительность при использовании регулярных выражений?

Лабораторная работа 9. Основы объектно-ориентированного программирования в Python

Название: Основы объектно-ориентированного программирования в Python

Цель: Познакомить студентов с основными концепциями объектно-ориентированного программирования, включая создание классов, объектов, и использование конструкторов и деструкторов в Python.

Задания:

1. **Основы ООП и создание классов**
 - Определите класс `Car` с атрибутами, такими как `make`, `model`, и `year`.
 - Создайте методы для вывода полной информации о машине и для изменения модели и года выпуска.
2. **Создание объектов**
 - Создайте несколько объектов класса `Car` с различными характеристиками.

- Вызовите методы объектов для вывода информации о каждой машине и измените атрибуты некоторых машин.

3. Конструкторы

- Реализуйте конструктор `__init__` в классе `Car`, который инициализирует все атрибуты класса при создании нового объекта.

- Создайте объекты, используя этот конструктор, и проверьте корректность инициализации.

4. Деструкторы

- Определите деструктор `__del__`, который будет выводить сообщение при удалении объекта.

- Создайте и удалите объект, наблюдая за работой деструктора.

5. Понимание концепций ООП

- Расширьте класс `Car`, добавив методы, которые позволяют взаимодействовать с другими объектами, например, метод для симуляции гонки между двумя автомобилями.

Контрольные вопросы:

1. Что такое объектно-ориентированное программирование и какие преимущества оно предоставляет?
2. Как определяются классы и объекты в Python?
3. Какова роль конструктора и деструктора в жизненном цикле объекта?
4. Как можно использовать классы для моделирования реальных объектов и процессов?
5. Какие общие практики и принципы следует учитывать при проектировании классов?

Лабораторная работа 10. Изучение атрибутов, свойств и методов в объектно-ориентированном программировании Python

Название: Изучение атрибутов, свойств и методов в объектно-ориентированном программировании Python

Цель: Научить студентов различать атрибуты классов и экземпляров, использовать геттеры и сеттеры с декоратором `property`, а также применять статические методы и методы класса для управления и обработки данных класса.

Задания:

1. Атрибуты классов и экземпляров

- Создайте класс `Person` с атрибутами класса, определяющими общее количество людей (счетчик экземпляров), и атрибутами экземпляра для имени и возраста.

- Создайте несколько экземпляров класса Person и выведите общее количество людей и атрибуты каждого экземпляра.

2. Геттеры, сеттеры и декоратор property

- Добавьте в класс Person свойство, которое позволяет получить возраст в днях (геттер), и установить возраст в годах (сеттер), используя декоратор property.

- Протестируйте геттер и сеттер, создав экземпляр класса и изменяя его возраст.

3. Статические методы и методы класса

- Определите в классе Person статический метод, который принимает дату рождения и текущую дату, и возвращает возраст.

- Определите метод класса, который возвращает количество созданных экземпляров класса.

- Продемонстрируйте использование этих методов, вызывая их на классе и на экземплярах.

4. Продвинутое задание: Использование всех типов методов

- Добавьте в класс Person метод экземпляра, который выводит всю доступную информацию о человеке.

- Используйте все созданные методы и свойства для работы с несколькими экземплярами класса.

Контрольные вопросы:

1. В чем разница между атрибутами класса и атрибутами экземпляра?
2. Как работают геттеры и сеттеры в Python и для чего они нужны?
3. Что такое декоратор property и как он используется?
4. Какие функции выполняют статические методы и методы класса? В чем различие между ними?
5. Как можно использовать статические методы и методы класса для управления данными в приложении?

Лабораторная работа 11. Магические методы в Python

Название: Магические методы в Python

Цель: Изучить основные магические методы в Python, научиться использовать их для настройки поведения классов при создании экземпляров, форматировании вывода, выполнении операций сравнения и работы с контейнерами.

Задания:

1. **Использование методов `__init__`, `__str__`, и `__repr__`**

- Создайте класс Book с атрибутами для названия, автора и года издания.
- Определите методы `__init__` для инициализации атрибутов, `__str__` для представления объекта в виде строки, пригодной для пользователя, и `__repr__` для официального строкового представления объекта.

2. Методы для операций сравнения

- Добавьте в класс Book магические методы `__eq__` (равенство), `__lt__` (меньше) и `__gt__` (больше), позволяющие сравнивать книги по году издания.
- Создайте несколько экземпляров книг и продемонстрируйте сравнение объектов.

3. Магические методы для контейнеров

- Расширьте класс BookShelf (книжная полка), который будет хранить объекты класса Book.
- Имплементируйте методы `__len__` для возвращения количества книг на полке и `__getitem__` для доступа к книгам по их индексу.
- Продемонстрируйте добавление книг в BookShelf и доступ к ним через индексацию.

Контрольные вопросы:

1. Что такое магические методы в Python и как они используются?
2. Какие магические методы используются для форматирования вывода объектов и в чем различие между `__str__` и `__repr__`?
3. Как реализовать и использовать магические методы для сравнения объектов?
4. Какие магические методы применимы для работы с контейнерными классами и какова их роль?
5. Приведите примеры, когда использование магических методов может значительно улучшить читаемость и функциональность кода.

Лабораторная работа 12. Наследование, инкапсуляция и полиморфизм в Python

Название: Наследование, инкапсуляция и полиморфизм в Python

Цель: Познакомить студентов с основами наследования, инкапсуляции и полиморфизма в Python, научить их применять эти принципы для создания эффективных и гибких объектно-ориентированных программ.

Задания:

1. **Основы наследования**
 - Создайте базовый класс Vehicle с атрибутами, общими для всех транспортных средств (например, `make`, `model`, `year`).

- Создайте производные классы Car и Truck, наследующие от Vehicle, и добавьте специфические атрибуты и методы (например, cargo_size для Truck).

2. Множественное наследование

- Создайте классы Electric и Manual, определяющие тип привода.
- Создайте класс ElectricTruck, который будет наследовать от Truck и Electric.

Продемонстрируйте работу атрибутов и методов, унаследованных от обоих классов.

3. Инкапсуляция данных

- Добавьте в класс Car приватные атрибуты (например, _engine_type) и публичные методы для доступа к этим атрибутам (геттеры и сеттеры).

4. Полиморфизм

- Определите в классе Vehicle метод display_info(), который выводит информацию о транспортном средстве.
 - Переопределите метод display_info() в классах Car и Truck так, чтобы он дополнительно выводил специфическую информацию для каждого типа транспортных средств.
 - Создайте объекты классов Car и Truck и вызовите метод display_info() для каждого из них, демонстрируя полиморфизм.

Контрольные вопросы:

1. Что такое наследование в контексте объектно-ориентированного программирования и каковы его преимущества?
2. Как реализовать множественное наследование в Python и какие могут быть при этом трудности?
3. Каковы принципы и преимущества инкапсуляции данных в ООП?
4. Что такое полиморфизм и как его можно реализовать в Python?
5. Какие реальные задачи можно решать с помощью наследования, инкапсуляции и полиморфизма в программировании?

Лабораторная работа 13. Работа с модулями и пакетами в Python

Название: Работа с модулями и пакетами в Python

Цель: Познакомить студентов с системой модулей и пакетов в Python, научить их создавать, структурировать и использовать модули и пакеты для организации кода и повторного использования функционала.

Задания:

1. **Импорт стандартных модулей**

- Импортируйте модуль `math` и используйте его функции для выполнения различных математических операций, таких как возведение в степень, вычисление корня, тригонометрические функции.
- Продемонстрируйте использование модуля `datetime` для работы с датами и временем.
- 2. **Создание и использование пользовательского модуля**
- Создайте модуль `utils.py`, который содержит функции для обработки строк (например, функцию для обратного порядка строки и функцию для проверки, является ли строка палиндромом).
- Импортируйте и используйте функции из `utils.py` в другом Python скрипте.
- 3. **Структурирование модулей в пакет**
- Создайте пакет `data_processing`, который включает модули для обработки данных, такие как `cleaning.py` и `analysis.py`.
- Демонстрируйте использование функций из этих модулей в основном скрипте.
- 4. **Пространства имен и относительные импорты**
- Внутри пакета `data_processing` используйте относительные импорты для доступа к функциям из одного модуля в другом.
- Объясните, как пространства имен помогают избежать конфликтов имен в больших проектах.

Контрольные вопросы:

1. Что такое модуль в Python и какие задачи он помогает решать?
2. Какие есть способы импорта модулей и какие из них предпочтительнее в различных ситуациях?
3. Что такое пакет и каковы преимущества его использования при структурировании кода?
4. Как работают пространства имен в Python и как они влияют на доступность модулей и пакетов?
5. Приведите примеры, когда использование пользовательских модулей и пакетов может существенно упростить разработку и поддержку программного обеспечения.

КОМПЛЕКТ ОЦЕНОЧНЫХ СРЕДСТВ К РУБЕЖНОМУ КОНТРОЛЮ

Первая рубежная аттестация (4-й семестр)

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 1

1. Объясните, как в Python используются переменные и какие правила следует соблюдать при их объявлении.
2. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 2

1. Опишите, какие операции можно проводить над различными типами данных и как они влияют на результаты программы.
2. Обсудите процесс и необходимость преобразования типов в Python, приведя примеры кода, где это преобразование критично.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 3

1. Опишите, как развивался Python с момента его создания до наших дней, и укажите основные этапы его эволюции.
2. Подробно опишите, как работают условные конструкции в Python и в чем их важность при создании алгоритмов.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 4

1. Объясните, как осуществляется индексация и работа со срезами в списках Python, и как эти возможности можно использовать для обработки данных.
2. Опишите, какие операции можно проводить над различными типами данных и как они влияют на результаты программы.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 5

1. Разъясните использование циклов for и while в Python на примерах кода, объясняя, в каких случаях предпочтительнее использовать каждый из них.
2. Сравните типы данных в Python (целые числа, вещественные числа, строки, булевы значения) с типами данных в других языках программирования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 6

1. Подробно опишите, как работают условные конструкции в Python и в чем их важность при создании алгоритмов.
2. Разъясните использование циклов for и while в Python на примерах кода, объясняя, в каких случаях предпочтительнее использовать каждый из них.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 7

1. Объясните, как осуществляется индексация и работа со срезами в списках Python, и как эти возможности можно использовать для обработки данных.
2. Опишите, как развивался Python с момента его создания до наших дней, и укажите основные этапы его эволюции.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 8

1. Сравните типы данных в Python (целые числа, вещественные числа, строки, булевы значения) с типами данных в других языках программирования.
2. Обсудите роль операторов break, continue и else в управлении циклами в Python, приведя примеры, когда их применение оправдано.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 9

1. Сравните типы данных в Python (целые числа, вещественные числа, строки, булевы значения) с типами данных в других языках программирования.
2. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 10

1. Объясните, как в Python используются переменные и какие правила следует соблюдать при их объявлении.
2. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 11

1. Рассмотрите методы и функции для работы со списками в Python, объяснив, как каждый метод изменяет данные списка.
2. Обсудите роль операторов break, continue и else в управлении циклами в Python, приведя примеры, когда их применение оправдано.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 12

1. Объясните, как осуществляется индексация и работа со срезами в списках Python, и как эти возможности можно использовать для обработки данных.
2. Сравните типы данных в Python (целые числа, вещественные числа, строки, булевы значения) с типами данных в других языках программирования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 13

1. Опишите, какие операции можно проводить над различными типами данных и как они влияют на результаты программы.
2. Обсудите процесс и необходимость преобразования типов в Python, приведя примеры кода, где это преобразование критично.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"

Билет № 14

1. Опишите, как развивался Python с момента его создания до наших дней, и укажите основные этапы его эволюции.
2. Опишите, какие операции можно проводить над различными типами данных и как они влияют на результаты программы.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 15

1. Обсудите роль операторов break, continue и else в управлении циклами в Python, приведя примеры, когда их применение оправдано.
2. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 16

1. Разъясните использование циклов for и while в Python на примерах кода, объясняя, в каких случаях предпочтительнее использовать каждый из них.
2. Опишите процесс создания и использования списков и кортежей в Python, указав ключевые различия и сценарии их применения.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 17

1. Разъясните использование циклов for и while в Python на примерах кода, объясняя, в каких случаях предпочтительнее использовать каждый из них.
2. Обсудите процесс и необходимость преобразования типов в Python, приведя примеры кода, где это преобразование критично.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 18

1. Опишите, какие операции можно проводить над различными типами данных и как они влияют на результаты программы.
2. Опишите, как развивался Python с момента его создания до наших дней, и укажите основные этапы его эволюции.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 19

1. Опишите процесс создания и использования списков и кортежей в Python, указав ключевые различия и сценарии их применения.
2. Сравните типы данных в Python (целые числа, вещественные числа, строки, булевы значения) с типами данных в других языках программирования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 20

1. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.
2. Опишите процесс создания и использования списков и кортежей в Python, указав ключевые различия и сценарии их применения.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 1

1. Объясните, что такое итераторы в Python, и как они используются для управления потоками данных.
2. Изложите, как и почему используются анонимные функции (лямбды) в Python, приведя примеры их применения для решения программных задач.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 2

1. Обсудите реальные сценарии использования регулярных выражений в программировании, демонстрируя, как они могут упростить и ускорить обработку данных.
2. Каковы основы определения и вызова функций в Python? Проиллюстрируйте свой ответ примерами, подчеркивая важность правильного использования функций в программировании.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 3

1. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.
2. Проанализируйте концепцию областей видимости переменных в Python, объясняя, как локальная и глобальная области видимости влияют на данные.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 4

1. Проанализируйте функции модуля `re`, такие как поиск, замена и разделение строк, с примерами их использования.
2. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 5

1. Рассмотрите использование менеджера контекста `with` при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.
2. Проанализируйте функции модуля `re`, такие как поиск, замена и разделение строк, с примерами их использования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 6

1. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.
2. Объясните методы чтения и записи данных в файлы, включая примеры кода, которые иллюстрируют эти процессы.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 7

1. Опишите основы синтаксиса регулярных выражений и как они применяются для обработки текстов в Python.

2. Проанализируйте концепцию областей видимости переменных в Python, объясняя, как локальная и глобальная области видимости влияют на данные.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 8

1. Объясните, что такое итераторы в Python, и как они используются для управления потоками данных.
2. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 9

1. Опишите процесс открытия и закрытия файлов в Python, подчеркивая важность правильного управления файловыми ресурсами.
2. Рассмотрите использование менеджера контекста with при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 10

1. Проанализируйте концепцию областей видимости переменных в Python, объясняя, как локальная и глобальная области видимости влияют на данные.
2. Объясните, как в Python работают параметры и аргументы функций, включая различия между позиционными и ключевыми аргументами.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 11

1. Изложите, как и почему используются анонимные функции (лямбды) в Python, приведя примеры их применения для решения программных задач.
2. Расскажите, как создаются и используются генераторы в Python, включая примеры функций и выражений, которые позволяют генерировать последовательности данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 12

1. Опишите процесс открытия и закрытия файлов в Python, подчеркивая важность правильного управления файловыми ресурсами.
2. Проанализируйте функции модуля re, такие как поиск, замена и разделение строк, с примерами их использования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 13

1. Расскажите о различных типах возвращаемых значений в функции, и как возвращаемые значения влияют на поток данных в программе.
2. Проанализируйте концепцию областей видимости переменных в Python, объясняя, как локальная и глобальная области видимости влияют на данные.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 14

1. Опишите основы синтаксиса регулярных выражений и как они применяются для обработки текстов в Python.
2. Объясните, что такое итераторы в Python, и как они используются для управления потоками данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 15

1. Каковы основы определения и вызова функций в Python? Проиллюстрируйте свой ответ примерами, подчеркивая важность правильного использования функций в программировании.
2. Изложите, как и почему используются анонимные функции (лямбды) в Python, приведя примеры их применения для решения программных задач.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 16

1. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.
2. Рассмотрите использование менеджера контекста with при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 17

1. Рассмотрите использование менеджера контекста with при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.
2. Расскажите о различных типах возвращаемых значений в функции, и как возвращаемые значения влияют на поток данных в программе.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 18

1. Опишите основы синтаксиса регулярных выражений и как они применяются для обработки текстов в Python.
2. Проанализируйте функции модуля `re`, такие как поиск, замена и разделение строк, с примерами их использования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 19

1. Проанализируйте концепцию областей видимости переменных в Python, объясняя, как локальная и глобальная области видимости влияют на данные.
2. Каковы основы определения и вызова функций в Python? Проиллюстрируйте свой ответ примерами, подчеркивая важность правильного использования функций в программировании.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 20

1. Расскажите, как создаются и используются генераторы в Python, включая примеры функций и выражений, которые позволяют генерировать последовательности данных.
2. Каковы основы определения и вызова функций в Python? Проиллюстрируйте свой ответ примерами, подчеркивая важность правильного использования функций в программировании.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"

Дисциплина "Разработка ПО на языке Python"

Билет № 1

1. Как классы и объекты реализуются в Python? Приведите примеры создания классов и их использования через объекты.
2. Как механизм наследования в Python помогает в повторном использовании кода? Приведите примеры с множественным наследованием.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"

Дисциплина "Разработка ПО на языке Python"

Билет № 2

1. Объясните, как работают конструкторы и деструкторы в классах Python, и какую роль они играют в управлении жизненным циклом объектов.
2. Расскажите о важности и применении атрибутов классов и экземпляров в Python, давая примеры из практической разработки.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"

Дисциплина "Разработка ПО на языке Python"

Билет № 3

1. Как механизм наследования в Python помогает в повторном использовании кода? Приведите примеры с множественным наследованием.
2. Опишите основные концепции объектно-ориентированного программирования и их значение в разработке современного программного обеспечения.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 4

1. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора `property` для создания управляемых свойств.
2. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 5

1. Расскажите о важности и применении атрибутов классов и экземпляров в Python, давая примеры из практической разработки.
2. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 6

1. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора `property` для создания управляемых свойств.
2. В чем заключается польза и методика создания пакетов в Python? Объясните процесс структурирования больших проектов с помощью пакетов и модулей.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 7

1. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора `property` для создания управляемых свойств.

2. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 8

1. Опишите основные концепции объектно-ориентированного программирования и их значение в разработке современного программного обеспечения.
2. В чем заключается польза и методика создания пакетов в Python? Объясните процесс структурирования больших проектов с помощью пакетов и модулей.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 9

1. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.
2. Как механизм наследования в Python помогает в повторном использовании кода? Приведите примеры с множественным наследованием.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 10

1. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора property для создания управляемых свойств.
2. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 11

1. Опишите различие и применение статических методов и методов класса. В чем их уникальность и когда они могут быть особенно полезны?
2. Как реализуется полиморфизм в Python? Дайте примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 12

1. Какие проблемы могут возникнуть при работе с наследованием и как Python позволяет их решать? Обсудите конкретные случаи, связанные с разрешением методов в иерархиях классов.
2. Как классы и объекты реализуются в Python? Приведите примеры создания классов и их использования через объекты.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 13

1. Какие проблемы могут возникнуть при работе с наследованием и как Python позволяет их решать? Обсудите конкретные случаи, связанные с разрешением методов в иерархиях классов.
2. Как реализуется полиморфизм в Python? Дайте примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"

Билет № 14

1. Опишите основные концепции объектно-ориентированного программирования и их значение в разработке современного программного обеспечения.
2. В чем заключается польза и методика создания пакетов в Python? Объясните процесс структурирования больших проектов с помощью пакетов и модулей.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 15

1. Как классы и объекты реализуются в Python? Приведите примеры создания классов и их использования через объекты.
2. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора property для создания управляемых свойств.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 16

1. Как реализуется полиморфизм в Python? Дайте примеры, показывающие, как один интерфейс может использоваться для различных типов данных.
2. Опишите различие и применение статических методов и методов класса. В чем их уникальность и когда они могут быть особенно полезны?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 17

1. Объясните, как работают конструкторы и деструкторы в классах Python, и какую роль они играют в управлении жизненным циклом объектов.
2. Опишите основные концепции объектно-ориентированного программирования и их значение в разработке современного программного обеспечения.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 18

1. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.
2. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 19

1. Расскажите о важности и применении атрибутов классов и экземпляров в Python, давая примеры из практической разработки.
2. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора property для создания управляемых свойств.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 20

1. Как механизм наследования в Python помогает в повторном использовании кода? Приведите примеры с множественным наследованием.
2. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 1

1. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?
2. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 2

1. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?
2. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 3

1. Обсудите преимущества и недостатки использования полиморфизма в объектно-ориентированном программировании. Как полиморфизм может улучшить или усложнить код в больших программных проектах?
2. Как магические методы влияют на поведение объектов при взаимодействии с Python системами, такими как циклы, функции и даже с самим интерпретатором Python? Приведите примеры использования магических методов в реальных приложениях.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 4

1. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?
2. Объясните, как реализуется полиморфизм в Python. Приведите примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 5

1. Объясните, как магические методы используются для работы с контейнерами в Python. Приведите примеры методов `__len__` и `__getitem__` и объясните, как они влияют на поведение классов, реализующих контейнеры.
2. Опишите, как магические методы `__init__`, `__str__`, и `__repr__` используются в Python для управления поведением объектов при создании, преобразовании в строку и официальном представлении объектов. Приведите примеры, иллюстрирующие их использование.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 6

1. Обсудите, как пакеты и пространства имен помогают управлять большими программными проектами в Python. Приведите примеры, как структурирование кода в пакеты улучшает модульность и предотвращает конфликты имен.
2. Объясните, как магические методы используются для работы с контейнерами в Python. Приведите примеры методов `__len__` и `__getitem__` и объясните, как они влияют на поведение классов, реализующих контейнеры.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"

Билет № 7

1. Объясните, как магические методы используются для работы с контейнерами в Python. Приведите примеры методов `__len__` и `__getitem__` и объясните, как они влияют на поведение классов, реализующих контейнеры.
2. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 8

1. Опишите процесс импорта стандартных и пользовательских модулей в Python. Какие проблемы могут возникнуть при неправильном управлении импортами?
2. Обсудите преимущества и недостатки использования полиморфизма в объектно-ориентированном программировании. Как полиморфизм может улучшить или усложнить код в больших программных проектах?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 9

1. Как магические методы влияют на поведение объектов при взаимодействии с Python системами, такими как циклы, функции и даже с самим интерпретатором Python? Приведите примеры использования магических методов в реальных приложениях.
2. Объясните, как реализуется полиморфизм в Python. Приведите примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 10

1. Объясните, как реализуется полиморфизм в Python. Приведите примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

2. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 11

1. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линейаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?
2. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 12

1. Разъясните, как реализовано множественное наследование в Python и какие вызовы могут возникнуть при его использовании. Приведите примеры кода.
2. Расскажите о роли магических методов для операций сравнения в Python, таких как `__eq__`, `__lt__`, и `__gt__`. Как эти методы влияют на функциональность классов?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 13

1. Какие практические меры можно предпринять для оптимизации структуры и производительности Python-программ, используя модули и пакеты? Приведите примеры, как организация кода в модули может помочь в оптимизации и упрощении сопровождения кода.
2. Обсудите принципы наследования в объектно-ориентированном программировании с примерами из Python. Как наследование способствует повторному использованию кода?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"

Дисциплина "Разработка ПО на языке Python"

Билет № 14

1. Какие практические меры можно предпринять для оптимизации структуры и производительности Python-программ, используя модули и пакеты? Приведите примеры, как организация кода в модули может помочь в оптимизации и упрощении сопровождения кода.
2. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"

Дисциплина "Разработка ПО на языке Python"

Билет № 15

1. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?
2. Опишите, как магические методы `__init__`, `__str__`, и `__repr__` используются в Python для управления поведением объектов при создании, преобразовании в строку и официальном представлении объектов. Приведите примеры, иллюстрирующие их использование.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"

Дисциплина "Разработка ПО на языке Python"

Билет № 16

1. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?
2. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линеаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 17

1. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линеаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?
2. Обсудите, как пакеты и пространства имен помогают управлять большими программными проектами в Python. Приведите примеры, как структурирование кода в пакеты улучшает модульность и предотвращает конфликты имен.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 18

1. Объясните, как магические методы используются для работы с контейнерами в Python. Приведите примеры методов `__len__` и `__getitem__` и объясните, как они влияют на поведение классов, реализующих контейнеры.
2. Обсудите принципы наследования в объектно-ориентированном программировании с примерами из Python. Как наследование способствует повторному использованию кода?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 19

1. Разъясните, как реализовано множественное наследование в Python и какие вызовы могут возникнуть при его использовании. Приведите примеры кода.
2. Опишите, как магические методы `__init__`, `__str__`, и `__repr__` используются в Python для управления поведением объектов при создании, преобразовании в строку и официальном представлении объектов. Приведите примеры, иллюстрирующие их использование.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 20

1. Объясните, как магические методы используются для работы с контейнерами в Python. Приведите примеры методов `__len__` и `__getitem__` и объясните, как они влияют на поведение классов, реализующих контейнеры.
2. Объясните, как реализуется полиморфизм в Python. Приведите примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

Подпись преподавателя _____ **Подпись заведующего кафедрой** _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "4"

Дисциплина "Разработка ПО на языке Python"

Билет № 1

1. Обсудите реальные сценарии использования регулярных выражений в программировании, демонстрируя, как они могут упростить и ускорить обработку данных.
2. Расскажите, как создаются и используются генераторы в Python, включая примеры функций и выражений, которые позволяют генерировать последовательности данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "4"

Дисциплина "Разработка ПО на языке Python"

Билет № 2

1. Проанализируйте функции модуля re, такие как поиск, замена и разделение строк, с примерами их использования.
2. Объясните, что такое итераторы в Python, и как они используются для управления потоками данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "4"

Дисциплина "Разработка ПО на языке Python"

Билет № 3

1. Опишите процесс открытия и закрытия файлов в Python, подчеркивая важность правильного управления файловыми ресурсами.
2. Обсудите роль операторов break, continue и else в управлении циклами в Python, приведя примеры, когда их применение оправдано.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 4

1. Рассмотрите методы и функции для работы со списками в Python, объяснив, как каждый метод изменяет данные списка.
2. Объясните, как осуществляется индексация и работа со срезами в списках Python, и как эти возможности можно использовать для обработки данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 5

1. Опишите основы синтаксиса регулярных выражений и как они применяются для обработки текстов в Python.
2. Объясните, как в Python работают параметры и аргументы функций, включая различия между позиционными и ключевыми аргументами.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 6

1. Обсудите реальные сценарии использования регулярных выражений в программировании, демонстрируя, как они могут упростить и ускорить обработку данных.
2. Проанализируйте функции модуля re, такие как поиск, замена и разделение строк, с примерами их использования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 7

1. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.

2. Объясните методы чтения и записи данных в файлы, включая примеры кода, которые иллюстрируют эти процессы.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 8

1. Опишите процесс открытия и закрытия файлов в Python, подчеркивая важность правильного управления файловыми ресурсами.
2. Обсудите реальные сценарии использования регулярных выражений в программировании, демонстрируя, как они могут упростить и ускорить обработку данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 9

1. Опишите процесс открытия и закрытия файлов в Python, подчеркивая важность правильного управления файловыми ресурсами.
2. Объясните, как в Python работают параметры и аргументы функций, включая различия между позиционными и ключевыми аргументами.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 10

1. Опишите процесс открытия и закрытия файлов в Python, подчеркивая важность правильного управления файловыми ресурсами.
2. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 11

1. Объясните, как в Python используются переменные и какие правила следует соблюдать при их объявлении.
2. Опишите основы синтаксиса регулярных выражений и как они применяются для обработки текстов в Python.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 12

1. Опишите процесс создания и использования списков и кортежей в Python, указав ключевые различия и сценарии их применения.
2. Опишите, как развивался Python с момента его создания до наших дней, и укажите основные этапы его эволюции.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 13

1. Объясните, как в Python работают параметры и аргументы функций, включая различия между позиционными и ключевыми аргументами.
2. Расскажите о различных типах возвращаемых значений в функции, и как возвращаемые значения влияют на поток данных в программе.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 14

1. Обсудите реальные сценарии использования регулярных выражений в программировании, демонстрируя, как они могут упростить и ускорить обработку данных.
2. Расскажите о различных типах возвращаемых значений в функциях, и как возвращаемые значения влияют на поток данных в программе.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 15

1. Проанализируйте функции модуля `re`, такие как поиск, замена и разделение строк, с примерами их использования.
2. Объясните, как в Python работают параметры и аргументы функций, включая различия между позиционными и ключевыми аргументами.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 16

1. Изложите, как и почему используются анонимные функции (лямбды) в Python, приведя примеры их применения для решения программных задач.
2. Проанализируйте концепцию областей видимости переменных в Python, объясняя, как локальная и глобальная области видимости влияют на данные.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 17

1. Проанализируйте функции модуля `re`, такие как поиск, замена и разделение строк, с примерами их использования.
2. Рассмотрите использование менеджера контекста `with` при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "4"

Дисциплина "Разработка ПО на языке Python"

Билет № 18

1. Проанализируйте функции модуля `re`, такие как поиск, замена и разделение строк, с примерами их использования.
2. Обсудите процесс и необходимость преобразования типов в Python, приведя примеры кода, где это преобразование критично.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "4"

Дисциплина "Разработка ПО на языке Python"

Билет № 19

1. Каковы основы определения и вызова функций в Python? Проиллюстрируйте свой ответ примерами, подчеркивая важность правильного использования функций в программировании.
2. Объясните методы чтения и записи данных в файлы, включая примеры кода, которые иллюстрируют эти процессы.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "4"

Дисциплина "Разработка ПО на языке Python"

Билет № 20

1. Рассмотрите использование менеджера контекста `with` при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.
2. Проанализируйте функции модуля `re`, такие как поиск, замена и разделение строк, с примерами их использования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 21

1. Опишите, какие операции можно проводить над различными типами данных и как они влияют на результаты программы.
2. Объясните, как осуществляется индексация и работа со срезами в списках Python, и как эти возможности можно использовать для обработки данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 22

1. Рассмотрите методы и функции для работы со списками в Python, объяснив, как каждый метод изменяет данные списка.
2. Объясните, как в Python работают параметры и аргументы функций, включая различия между позиционными и ключевыми аргументами.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 23

1. Опишите процесс открытия и закрытия файлов в Python, подчеркивая важность правильного управления файловыми ресурсами.
2. Подробно опишите, как работают условные конструкции в Python и в чем их важность при создании алгоритмов.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 24

1. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.

2. Сравните типы данных в Python (целые числа, вещественные числа, строки, булевы значения) с типами данных в других языках программирования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 25

1. Обсудите процесс и необходимость преобразования типов в Python, приведя примеры кода, где это преобразование критично.
2. Расскажите, как создаются и используются генераторы в Python, включая примеры функций и выражений, которые позволяют генерировать последовательности данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 26

1. Объясните, что такое итераторы в Python, и как они используются для управления потоками данных.
2. Объясните, как в Python используются переменные и какие правила следует соблюдать при их объявлении.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 27

1. Расскажите, как создаются и используются генераторы в Python, включая примеры функций и выражений, которые позволяют генерировать последовательности данных.
2. Объясните, как в Python используются переменные и какие правила следует соблюдать при их объявлении.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 28

1. Рассмотрите использование менеджера контекста with при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.
2. Разъясните использование циклов for и while в Python на примерах кода, объясняя, в каких случаях предпочтительнее использовать каждый из них.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 29

1. Изложите, как и почему используются анонимные функции (лямбды) в Python, приведя примеры их применения для решения программных задач.
2. Опишите, какие операции можно проводить над различными типами данных и как они влияют на результаты программы.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 30

1. Объясните, что такое итераторы в Python, и как они используются для управления потоками данных.
2. Объясните, как в Python работают параметры и аргументы функций, включая различия между позиционными и ключевыми аргументами.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 31

1. Обсудите процесс и необходимость преобразования типов в Python, приведя примеры кода, где это преобразование критично.
2. Опишите, как развивался Python с момента его создания до наших дней, и укажите основные этапы его эволюции.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 32

1. Опишите основы синтаксиса регулярных выражений и как они применяются для обработки текстов в Python.
2. Подробно опишите, как работают условные конструкции в Python и в чем их важность при создании алгоритмов.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 33

1. Расскажите о различных типах возвращаемых значений в функции, и как возвращаемые значения влияют на поток данных в программе.
2. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 34

1. Опишите основы синтаксиса регулярных выражений и как они применяются для обработки текстов в Python.
2. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 35

1. Проанализируйте концепцию областей видимости переменных в Python, объясняя, как локальная и глобальная области видимости влияют на данные.
2. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 36

1. Рассмотрите использование менеджера контекста with при работе с файлами, объяснив, как он помогает управлять файловыми потоками и предотвращать ошибки.
2. Объясните, что такое программирование и почему оно играет критически важную роль в современном мире.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 37

1. Опишите преимущества использования генераторов в Python, включая сценарии, где их использование оптимально с точки зрения производительности и памяти.
2. Проанализируйте функции модуля re, такие как поиск, замена и разделение строк, с примерами их использования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 38

1. Объясните, что такое итераторы в Python, и как они используются для управления потоками данных.

2. Проанализируйте функции модуля re, такие как поиск, замена и разделение строк, с примерами их использования.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 39

1. Опишите, как развивался Python с момента его создания до наших дней, и укажите основные этапы его эволюции.
2. Подробно опишите, как работают условные конструкции в Python и в чем их важность при создании алгоритмов.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "4"
Дисциплина "Разработка ПО на языке Python"
Билет № 40

1. Обсудите процесс и необходимость преобразования типов в Python, приведя примеры кода, где это преобразование критично.
2. Расскажите о различных типах возвращаемых значений в функциях, и как возвращаемые значения влияют на поток данных в программе.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Вопросы к экзамену (5-й семестр)

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 1

1. Обсудите принципы наследования в объектно-ориентированном программировании с примерами из Python. Как наследование способствует повторному использованию кода?

2. Обсудите преимущества и недостатки использования полиморфизма в объектно-ориентированном программировании. Как полиморфизм может улучшить или усложнить код в больших программных проектах?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 2

1. В чем заключается польза и методика создания пакетов в Python? Объясните процесс структурирования больших проектов с помощью пакетов и модулей.
2. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм СЗ линеаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 3

1. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.
2. Как реализуется полиморфизм в Python? Дайте примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 4

1. Как реализуется полиморфизм в Python? Дайте примеры, показывающие, как один интерфейс может использоваться для различных типов данных.
2. Как механизм наследования в Python помогает в повторном использовании кода? Приведите примеры с множественным наследованием.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"

Дисциплина "Разработка ПО на языке Python"

Билет № 5

1. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линеаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?
2. Объясните, как реализуется полиморфизм в Python. Приведите примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"

Дисциплина "Разработка ПО на языке Python"

Билет № 6

1. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?
2. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора `property` для создания управляемых свойств.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"

Дисциплина "Разработка ПО на языке Python"

Билет № 7

1. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линеаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?
2. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 8

1. Обсудите, как пакеты и пространства имен помогают управлять большими программными проектами в Python. Приведите примеры, как структурирование кода в пакеты улучшает модульность и предотвращает конфликты имен.
2. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 9

1. Обсудите преимущества и недостатки использования полиморфизма в объектно-ориентированном программировании. Как полиморфизм может улучшить или усложнить код в больших программных проектах?
2. Какие практические меры можно предпринять для оптимизации структуры и производительности Python-программ, используя модули и пакеты? Приведите примеры, как организация кода в модули может помочь в оптимизации и упрощении сопровождения кода.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 10

1. Опишите основные концепции объектно-ориентированного программирования и их значение в разработке современного программного обеспечения.
2. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 11

1. Изложите концепции инкапсуляции данных в Python. Как можно защитить данные в классе от нежелательного доступа?
2. Опишите основные концепции объектно-ориентированного программирования и их значение в разработке современного программного обеспечения.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 12

1. Какие практические меры можно предпринять для оптимизации структуры и производительности Python-программ, используя модули и пакеты? Приведите примеры, как организация кода в модули может помочь в оптимизации и упрощении сопровождения кода.
2. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линеаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 13

1. Объясните, как магические методы используются для работы с контейнерами в Python. Приведите примеры методов `__len__` и `__getitem__` и объясните, как они влияют на поведение классов, реализующих контейнеры.
2. Обсудите, как пакеты и пространства имен помогают управлять большими программными проектами в Python. Приведите примеры, как структурирование кода в пакеты улучшает модульность и предотвращает конфликты имен.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 14

1. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?

2. Обсудите преимущества и недостатки использования полиморфизма в объектно-ориентированном программировании. Как полиморфизм может улучшить или усложнить код в больших программных проектах?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 15

1. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?
2. Объясните, как работают конструкторы и деструкторы в классах Python, и какую роль они играют в управлении жизненным циклом объектов.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 16

1. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?
2. Опишите различие и применение статических методов и методов класса. В чем их уникальность и когда они могут быть особенно полезны?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 17

1. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?
2. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора property для создания управляемых свойств.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 18

1. Опишите, как магические методы `__init__`, `__str__`, и `__repr__` используются в Python для управления поведением объектов при создании, преобразовании в строку и официальном представлении объектов. Приведите примеры, иллюстрирующие их использование.
2. Как реализуется полиморфизм в Python? Дайте примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 19

1. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.
2. Опишите различие и применение статических методов и методов класса. В чем их уникальность и когда они могут быть особенно полезны?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 20

1. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?
2. Расскажите о важности и применении атрибутов классов и экземпляров в Python, давая примеры из практической разработки.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 21

1. Расскажите о важности и применении атрибутов классов и экземпляров в Python, давая примеры из практической разработки.
2. Обсудите, как пакеты и пространства имен помогают управлять большими программными проектами в Python. Приведите примеры, как структурирование кода в пакеты улучшает модульность и предотвращает конфликты имен.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 22

1. Какие практические меры можно предпринять для оптимизации структуры и производительности Python-программ, используя модули и пакеты? Приведите примеры, как организация кода в модули может помочь в оптимизации и упрощении сопровождения кода.
2. Обсудите принципы наследования в объектно-ориентированном программировании с примерами из Python. Как наследование способствует повторному использованию кода?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 23

1. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора property для создания управляемых свойств.
2. Объясните, как реализуется полиморфизм в Python. Приведите примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 24

1. Объясните, как реализуется полиморфизм в Python. Приведите примеры, показывающие, как один интерфейс может использоваться для различных типов данных.
2. Объясните, как работают конструкторы и деструкторы в классах Python, и какую роль они играют в управлении жизненным циклом объектов.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 25

1. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.
2. Как магические методы влияют на поведение объектов при взаимодействии с Python системами, такими как циклы, функции и даже с самим интерпретатором Python? Приведите примеры использования магических методов в реальных приложениях.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 26

1. Обсудите, как пакеты и пространства имен помогают управлять большими программными проектами в Python. Приведите примеры, как структурирование кода в пакеты улучшает модульность и предотвращает конфликты имен.
2. Расскажите о важности и применении атрибутов классов и экземпляров в Python, давая примеры из практической разработки.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 27

1. Как реализуется полиморфизм в Python? Дайте примеры, показывающие, как один интерфейс может использоваться для различных типов данных.

2. Опишите, как магические методы `__init__`, `__str__`, и `__repr__` используются в Python для управления поведением объектов при создании, преобразовании в строку и официальном представлении объектов. Приведите примеры, иллюстрирующие их использование.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 28

1. Опишите основные концепции объектно-ориентированного программирования и их значение в разработке современного программного обеспечения.
2. Опишите процесс импорта стандартных и пользовательских модулей в Python. Какие проблемы могут возникнуть при неправильном управлении импортами?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 29

1. В чем заключается польза и методика создания пакетов в Python? Объясните процесс структурирования больших проектов с помощью пакетов и модулей.
2. Опишите процесс импорта стандартных и пользовательских модулей в Python. Какие проблемы могут возникнуть при неправильном управлении импортами?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 30

1. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?
2. В чем заключается польза и методика создания пакетов в Python? Объясните процесс структурирования больших проектов с помощью пакетов и модулей.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 31

1. Опишите процесс импорта стандартных и пользовательских модулей в Python. Какие проблемы могут возникнуть при неправильном управлении импортами?
2. Какие практические меры можно предпринять для оптимизации структуры и производительности Python-программ, используя модули и пакеты? Приведите примеры, как организация кода в модули может помочь в оптимизации и упрощении сопровождения кода.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 32

1. Опишите, как магические методы `__init__`, `__str__`, и `__repr__` используются в Python для управления поведением объектов при создании, преобразовании в строку и официальном представлении объектов. Приведите примеры, иллюстрирующие их использование.
2. Приведите примеры, как ООП применяется для решения типичных задач программирования. Какие паттерны проектирования особенно популярны в Python и почему?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 33

1. Как геттеры и сеттеры используются для управления доступом к данным в Python? Проиллюстрируйте использование декоратора `property` для создания управляемых свойств.
2. Опишите процесс импорта стандартных и пользовательских модулей в Python. Какие проблемы могут возникнуть при неправильном управлении импортами?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"

Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 34

1. Расскажите о создании и структурировании модулей в Python. Как можно организовать код, чтобы улучшить его читаемость и поддерживаемость?
2. Как классы и объекты реализуются в Python? Приведите примеры создания классов и их использования через объекты.

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 35

1. Объясните, как инкапсуляция данных влияет на безопасность и стабильность программы. Приведите примеры правильного использования приватных и защищенных атрибутов.
2. Опишите различие и применение статических методов и методов класса. В чем их уникальность и когда они могут быть особенно полезны?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 36

1. Как магические методы влияют на поведение объектов при взаимодействии с Python системами, такими как циклы, функции и даже с самим интерпретатором Python? Приведите примеры использования магических методов в реальных приложениях.
2. Расскажите о роли магических методов для операций сравнения в Python, таких как `__eq__`, `__lt__`, и `__gt__`. Как эти методы влияют на функциональность классов?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 37

1. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линеаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?
2. Опишите процесс импорта стандартных и пользовательских модулей в Python. Какие проблемы могут возникнуть при неправильном управлении импортами?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 38

1. Объясните, как работают конструкторы и деструкторы в классах Python, и какую роль они играют в управлении жизненным циклом объектов.
2. Расскажите о роли магических методов для операций сравнения в Python, таких как `__eq__`, `__lt__`, и `__gt__`. Как эти методы влияют на функциональность классов?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 39

1. Рассмотрите процесс решения конфликтов при множественном наследовании в Python, используя алгоритм C3 линеаризации. Как этот алгоритм помогает обеспечить консистентность и предсказуемость порядка разрешения методов?
2. Обсудите принципы наследования в объектно-ориентированном программировании с примерами из Python. Как наследование способствует повторному использованию кода?

Подпись преподавателя _____ Подпись заведующего кафедрой _____

Грозненский государственный нефтяной технический университет им.акад. М.Д. Миллионщикова
Институт "цифровой экономики и технологического предпринимательства"
Группа "ПИ-24" Семестр "5"
Дисциплина "Разработка ПО на языке Python"
Билет № 40

1. Какие практические меры можно предпринять для оптимизации структуры и производительности Python-программ, используя модули и пакеты? Приведите примеры, как организация кода в модули может помочь в оптимизации и упрощении сопровождения кода.

2. Расскажите о роли магических методов для операций сравнения в Python, таких как `__eq__`, `__lt__`, и `__gt__`. Как эти методы влияют на функциональность классов?

Подпись преподавателя _____ Подпись заведующего кафедрой _____
