Федеральное государственное бюджетное образовательное учреждение высшего образования
РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА И ГОСУДАРСТВЕННОЙ СЛУЖБЫ при ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ
Среднерусский институт управления – филиал
ФАКУЛЬТЕТ «Технологии, товароведение и сервис»
КАФЕДРА «Математика и информационные технологии»

Меркулов Павел Александрович

Подписан: Меркулов Павел Александрович
DN: C=RU, S=Орловская область, L=пр. Орел, T=Директор, OU="СРЕДНЕРУССКИЙ ИНСТИТУТ УПРАВЛЕНИЯ - ФИЛИАЛ ФЕДЕРАЛЬНОГО ГОСУДАРСТВЕННОГО БЮДЖЕТНОГО ОБРАЗОВАТЕЛЬНОГО УЧРЕЖДЕНИЯ ВЫСШЕГО ОБРАЗОВАНИЯ ""РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА И ГОСУДАРСТВЕННОЙ СЛУЖБЫ ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ""", O="ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ ""РОССИЙСКАЯ АКАДЕМИЯ НАРОДНОГО ХОЗЯЙСТВА И ГОСУДАРСТВЕННОЙ СЛУЖБЫ ПРИ ПРЕЗИДЕНТЕ РОССИЙСКОЙ ФЕДЕРАЦИИ""", СНИЛС=00482216008, ИНН=575206486423, E=it@orel.ranepa.ru, G=Павел Александрович, SN=Меркулов, CN=Меркулов Павел Александрович
Основание: я подтверждаю этот документ своей удостоверяющей подписью
Местоположение: Среднерусский институт управления

УТВЕРЖДЕНО
Директор Среднерусского института управления – филиала РАНХ и ГС
П.А. Меркулов

ПРОГРАММА БАКАЛАВРИАТА

ИНФОРМАЦИОННЫЕ СИСТЕМЫ И ТЕХНОЛОГИИ В БИЗНЕСЕ

(наименование образовательной программы)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ,
реализуемой без применения электронного (онлайн) курса

Б1.О.19 Языки и технологии программирования

(код и наименование РПД)

09.03.02 Информационные системы и технологии

(код и наименование направления подготовки/специальности)

очная, очно-заочная, заочная

(форма (формы) обучения)

Год набора: 2022

Орел, 2022 г.

Автор–составитель:
к.п.н., доцент кафедры математики и информационных технологий А.Г. Савина

И.о. заведующего кафедрой
математики и информационных технологий, доктор экономических наук, профессор Л.И. Малявкина

РПД Б1.О.19 «Языки и технологии программирования» одобрена на заседании кафедры математики и информационных технологий, протокол от 26 августа 2022 г., №1.

СОДЕРЖАНИЕ

1. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы 4

2. Объем и место дисциплины в структуре образовательной программы 4

3. Содержание и структура дисциплины 5

4. Материалы текущего контроля успеваемости обучающихся 10

5. Оценочные материалы промежуточной аттестации по дисциплине 14

6. Методические указания для освоения дисциплины 16

7. Учебная литература и ресурсы информационно-телекоммуникационной сети Интернет 18

8. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы 19

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

1.1 Дисциплина Б1.О.19 «Языки и технологии программирования» обеспечивает овладение следующими компетенциями:

Код компетенции	Наименование компетенции	Код компонента компетенции	Наименование компонента компетенции
ОПК-6	Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	ОПК-6.2	Способен разрабатывать программы, пригодные для практического применения в области информационных систем и технологий

1.2 В результате освоения дисциплины у студентов должны быть сформированы:

ОТФ/ТФ/ трудовые или профессиональные действия	Код компонента компетенции	Результаты обучения
	ОПК-6.2	на уровне знаний: знает методы и этапы реализации технологии программирования, особенности структурного, функционального и объектно-ориентированного программирования
		на уровне умений: умеет выбирать методы программирования и разрабатывать программы с использованием структурного, функционального и объектно-ориентированного программирования
		на уровне навыков: демонстрирует навыки разработки программ с использованием структурного, функционального и объектно-ориентированного программирования

2. Объем и место дисциплины в структуре образовательной программы

Объем дисциплины

Общий объем дисциплины в зачетных единицах (академических/астрономических часах): 5 ЗЕ (180/135).

Количество академических/астрономических часов, выделенных на контактную работу:

– по очной форме обучения – 78/58,5 (в т.ч. лекц. – 26 ч., практ. – 52 ч.); на самостоятельную работу обучающихся – 75/56,25.

– по очно-заочной форме обучения – 24/18 (в т.ч. лекц. – 8 ч., практ. – 16 ч.); на самостоятельную работу обучающихся – 147/110,25.

– по заочной форме обучения – 24/18 (в т.ч. лекц. – 8 ч., практ. – 16 ч.); на самостоятельную работу обучающихся – 147/110,25.

Место дисциплины в структуре образовательной программы

Дисциплина относится к обязательной части, Блока 1. «Дисциплины (модули)» учебного плана образовательной программы.

Дисциплина изучается: очная форма обучения – 2 курс, 3 семестр; очно-заочная форма обучения – 3 курс, заочная форма обучения – 3 курс.

Освоение дисциплины опирается на содержание учебных дисциплин: Дискретная математика, Алгоритмизация и программирование

Основные положения дисциплины используются в дальнейшем при прохождении программирование и конфигурирование на платформе 1С:Предприятие, Проектирование и разработка баз данных.

Форма аттестации: экзамен

3. Содержание и структура дисциплины

3.1. Структура дисциплины

Очная форма обучения

№ п/п	Наименование тем и/или разделов	Объем дисциплины (модуля), час.						Форма текущего контроля успеваемости*, промежуточной аттестации**
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СРО	
			Л/ ДОТ	ЛР/ ДОТ	ПЗ/ ДОТ	КСР		
Тема 1	Общие принципы, основные подходы и этапы разработки программного обеспечения.	10	2		2		6	О
Тема 2	Эволюция и перспективы развития технологий программирования. Современные парадигмы программирования.	10	2		2		6	О
Тема 3	Концептуальная иерархия, базовые принципы и основные конструкции языка программирования Python. Структурное программирование	18	2		8		8	Т, ЗПР, К
Тема 4	Модули и пакеты Python. Работа с датой и временем в Python. Технологии работы с фай-	18	4		6		8	Т, ЗПР, К

	лами							
Тема 5	Событийно-ориентированное программирование. Программирование графического пользовательского интерфейса	20	4		6		10	Т, ЗПР, К
Тема 6	Функциональное программирование в Python 3	18	4		4		10	Т, ЗПР, К
Тема 7	Методология (парадигма) объектно-ориентированного программирования. Поддержка объектно- ориентированного программирования в Python	24	4		10		10	Т, ЗПР, К
Тема 8	Технологии анализа и визуализации данных с использованием библиотек Python	18	2		6		10	Т, ЗПР
Тема 9	Базовые инструменты фреймфорка Django для создания веб-сайта на Python	17	2		8		7	Т, ЗПР
Промежуточная аттестация		27						Экз
Всего:		180	26		52		75	

Примечание:
** формы заданий текущего контроля успеваемости: (контрольные работы (К), опрос (О), тестирование (Т), коллоквиум (Кол) и т.п.), защита практических работ (ЗПР), кейс-задания (КЗ) и виды учебных заданий (эссе (Эс), реферат (Реф), диспут (Д) и др.),*
** форма промежуточной аттестации: экзамен (Экз), зачет(З), зачет с оценкой (ЗО)*

Очно-заочная форма обучения

№ п/п	Наименование тем и/или разделов	Объем дисциплины (модуля), час.						Форма текущего контроля успеваемости*, промежуточной аттестации**
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СРО	
			Л/ ДОТ	ЛР/ ДОТ	ПЗ/ ДОТ	КСР		
Тема 1	Общие принципы, основные подходы и этапы разработки программного обеспечения.	11	1		-		10	О
Тема 2	Эволюция и пер-	19	1		2		16	О

	спективы развития технологий программирования. Современные парадигмы программирования.							
Тема 3	Концептуальная иерархия, базовые принципы и основные конструкции языка программирования Python. Структурное программирование	19	1		2		16	Т, ЗПР, К
Тема 4	Модули и пакеты Python. Работа с датой и временем в Python. Технологии работы с файлами	19	1		2		16	Т, ЗПР, К
Тема 5	Событийно-ориентированное программирование. Программирование графического пользовательского интерфейса	29	1		2		26	Т, ЗПР, К
Тема 6	Функциональное программирование в Python 3	23	1		2		20	Т, ЗПР, К
Тема 7	Методология (парадигма) объектно-ориентированного программирования. Поддержка объектно- ориентированного программирования в Python	19	1		2		16	Т, ЗПР, К
Тема 8	Технологии анализа и визуализации данных с использованием библиотек Python	13	1		2		10	Т, ЗПР
Тема 9	Базовые инструменты фреймфорка Django для создания веб-сайта на Python	19	-		2		17	Т, ЗПР

Промежуточная аттестация	9						Экз
Всего:	180	8		16		147	

Примечание:
** формы заданий текущего контроля успеваемости: (контрольные работы (К), опрос (О), тестирование (Т), коллоквиум (Кол) и т.п.) и виды учебных заданий (эссе (Эс), реферат (Реф), диспут (Д) и др.),*
** форма промежуточной аттестации: экзамен (Экз), зачет(З), зачет с оценкой (ЗО)*

Заочная форма обучения

№ п/п	Наименование тем и/или разделов	Объем дисциплины (модуля), час.						Форма текущего контроля успеваемости*, промежуточной аттестации**
		Всего	Контактная работа обучающихся с преподавателем по видам учебных занятий				СРО	
			Л/ ДОТ	ЛР/ ДОТ	ПЗ/ ДОТ	КСР		
Тема 1	Общие принципы, основные подходы и этапы разработки программного обеспечения.	11	1		-		10	О
Тема 2	Эволюция и перспективы развития технологий программирования. Современные парадигмы программирования.	19	1		2		16	О
Тема 3	Концептуальная иерархия, базовые принципы и основные конструкции языка программирования Python. Структурное программирование	19	1		2		16	Т, ЗПР, К
Тема 4	Модули и пакеты Python. Работа с датой и временем в Python. Технологии работы с файлами	19	1		2		16	Т, ЗПР, К
Тема 5	Событийно-ориентированное программирование. Программирование графического пользовательского интерфейса	29	1		2		26	Т, ЗПР, К
Тема 6	Функциональное программирование в Python 3	23	1		2		20	Т, ЗПР, К

Тема 7	Методология (парадигма) объектно-ориентированного программирования. Поддержка объектно- ориентированного программирования в Python	19	1		2		16	Т, ЗПР, К
Тема 8	Технологии анализа и визуализации данных с использованием библиотек Python	13	1		2		10	Т, ЗПР
Тема 9	Базовые инструменты фреймфорка Django для создания веб-сайта на Python	19	-		2		17	Т, ЗПР
Промежуточная аттестация		9						Экз
Всего:		180	8		16		147	

Примечание:
** формы заданий текущего контроля успеваемости: (контрольные работы (К), опрос (О), тестирование (Т), коллоквиум (Кол) и т.п.) и виды учебных заданий (эссе (Эс), реферат (Реф), диспут (Д) и др.),*
** форма промежуточной аттестации: экзамен (Экз), зачет(З), зачет с оценкой (ЗО)*

3.2.Содержание дисциплины

Тема 1. Общие принципы, основные подходы и этапы разработки программного обеспечения.

Основные подходы к разработке программного обеспечения. Этапы разработки. Модели жизненного цикла программного обеспечения. Стандарты в области разработки и реализации программного обеспечения. Методологические принципы разработки и реализации алгоритмов и программ.

Тема 2. Эволюция и перспективы развития технологий программирования. Современные парадигмы программирования

Современные представления о методах, технологиях и парадигмах программирования. Этапы эволюционного развития технологий программирования: неструктцрированное программирование, процедурное и модульное программирование, концепция объектно-ориентированного программирования, декларативное программирование. Компонентный подход в технологиях программирования. Обзор современных технологий программирования и тенденции их развития.

Тема 3. Концептуальная иерархия, базовые принципы и основные конструкции языка программирования Python. Структурное программирование

Концепция данных и основные конструкции языка программирования Python. Программирование линейных, разветвляющихся, циклических алгоритмов. Технологии работы со списками, строками, кортежами, множествами, словарями. Вложенные последовательности в Python.

Тема 4. Модули и пакеты Python. Работа с датой и временем в Python. Технологии работы с файлами

Модули и пакеты Python. Технологии подключения и работы с модулями random, string, turtle и др. Работа с датой и временем в Python. Технологии работы с файлами

Тема 5. Событийно-ориентированное программирование. Программирование графического пользовательского интерфейса

Понятие графического пользовательского интерфейса. GUI-библиотеки языка программирования Python. Архитектурные особенности модуля Tkinter. Технология визуального проектирования и событийно-ориентированного программирования. Этапы разработки GUI-приложения в модуле tkinter. Способы подключения модуля tkinter. Основные виджеты модуля tkinter и технологии работы с ними. Менеджеры размещения виджетов.

Тема 6. Функциональное программирование в Python 3

Понятие и виды функций в python. Встроенные функции. Технологии создания пользовательских функций и модулей.

Тема 7. Методология (парадигма) объектно-ориентированного программирования. Поддержка объектно- ориентированного программирования в Python

Основные принципы объектно-ориентированного программирования. Атрибуты объектов и классов. Методы экземпляра класса. Технологии работы с декораторами. Хеширование объектов. Протоколы: итерируемых объектов и итераторов, последовательностей, контекстных менеджеров. Дескрипторов. Наследование и полиморфизм.

Тема 8. Технологии анализа и визуализации данных с использованием библиотек Python

Базовые возможности языка Python для анализа данных. Обзор библиотек Python для анализа данных. Технологии анализа данных с использованием библиотеки Pandas.

Тема 9. Базовые инструменты фреймфорка Django для создания веб-сайта на Python

Установка Django. Структура проекта. Модели, представление, шаблоны. Расширение моделей и шаблонов: работа с изображениями, организация ссылок в шаблонах, расширение шаблонов. Создание формы заявки и ее улучшение. Фильтрация и поиск. Публикация сайта.

4. Материалы текущего контроля успеваемости обучающихся

4.1. В ходе реализации дисциплины Б1.О.19 «Языки и технологии программирования» используются следующие методы текущего контроля успеваемости обучающихся:

Тема и/или раздел	Методы текущего контроля успеваемости
Тема 1. Общие принципы, основные подходы и этапы разработки программного обеспечения.	Опрос
Тема 2. Эволюция и перспективы развития технологий программирования. Современные парадигмы программирования.	Опрос
Тема 3. Концептуальная иерархия, базовые принципы и основные конструкции языка программирования Python. Структурное программирование	Тестирование, защита практических работ, контрольная работа
Тема 4. Модули и пакеты Python. Работа с датой и временем в Python. Технологии работы с файлами	Тестирование, защита практических работ, контрольная работа
Тема 5. Событийно-ориентированное программирование. Программирование графического пользовательского интерфейса	Тестирование, защита практических работ, контрольная работа
Тема 6. Функциональное программирование в Python 3	Тестирование, защита практических работ, контрольная работа
Тема 7. Методология (парадигма) объектно-ориентированного программирования. Поддержка объектно-ориентированного программирования в Python	Тестирование, защита практических работ, контрольная работа

Тема 8. Технологии анализа и визуализации данных с использованием библиотек Python	Тестирование, защита практических работ
Тема 9. Базовые инструменты фреймфорка Django для создания веб-сайта на Python	Тестирование, защита практических работ

4.2. Типовые материалы текущего контроля успеваемости обучающихся

Примерные вопросы для опроса

1. Этапы эволюционного развития технологий программирования.
2. Какая модель построения программ лежит в основе процедурного программирования
3. Основные методы процедурного программирования
4. Особенности модульного программирование
5. Концепция объектно-ориентированного программирования Основные принципы объектно-ориентированного программирования
6. Тенденции развития технологий программирования

Примерные вопросы для защиты практических работ

7. Технологии работы в интегрированной среде разработки
8. Структура программы на языке Python, основные правила написания программного кода
9. Основные методы работы со списками (кортежами, множествами)
10. Основные методы работы со строками
11. Характеристика стадий тестирования. Основные принципы тестирования. Тесты и тестовые наборы. Понятие тестового покрытия.
12. Отладочное тестирование. Соотношение структурного и функционального подходов. Примеры реализации.
13. Технологии работы с датой и временем в Python.
14. Технологии работы с файлами в Python.
15. Технологии визуализации данных с использованием библиотеки Pandas
16. Базовые инструменты фреймфорка Django
17. Технологии разработки GUI-приложения на Python

Примерный вариант контрольной работы

Тема 3:

Задание 1. Напишите программу sort.py, которая получает из первого аргумента командной строки слово, а затем выводит все буквы слова в алфавитном порядке исключая повторные буквы.

Задание 2. Напишите программу count.py, которая получает из первого аргумента командной строки предложение, а затем выводит количество слов в этом предложении.

В предложении могут быть двойные пробелы, а также неправильно расставленные знаки препинания (запятые, точки, восклицательный и вопросительный знаки). Если в программу передать набор слов, заключенных в кавычки, то они будут восприняты как один аргумент.

Тема 4:

Список покупок включает три файла: shopping_list_1.txt, shopping_list_2.txt и shopping_ list_3.txt. Некоторые товары в файлах повторяются. Требуется составить нормализованный список, где продукты не повторяются и записаны по алфавиту. Напишите программу shopping_list.py, которая читает строки из трех файлов: а затем создает новый

файл shopping_list.txt, в который помещает все прочитанные строки без повторов и в алфавитном порядке. Примечание: все исходные списки заканчиваются пустой строкой.

Тема 6:

Задание 1. Часто для названия статьи на сайте или товара в интернет-магазине нужно сгенерировать уникальный идентификатор, который потом можно будет вставить в URL. Напишите функцию **gen_id**, которая генерирует такой идентификатор.

Первым аргументом функция принимает название товара на английском языке, а также список уже существующих идентификаторов. Новый идентификатор генерируется по следующим правилам:

1. Исходное имя преобразуется к нижнему регистру.
2. Пробелы справа и слева от названия удаляются.
3. Все пробелы заменяются на черточки.
4. Все символы, кроме символов латинского алфавита, цифр и знака «черточка», должны быть удалены.
5. Если полученный идентификатор уже существует, то функция должна прибавить в конец черточку и сгенерированное число, которое начинается с двойки. Если идентификатор с двойкой также существует, то число увеличивается на 1. И так до тех пор пока не будет найден идентификатор, которого не существует.

Задание 2. Напишите функцию capitalization, которая принимает три аргумента: вложенную сумму, годовой процент по вкладу и срок вклада в месяцах. Все три аргумента — целые числа. Функция должна возвращать сумму, которую получит пользователь по окончанию вклада. Возвращаемая сумма должна быть целым числом (предполагается ежемесячная капитализация).

Примерные вопросы для тестирования

1. Динамическая типизация языка Python предполагает связывание переменной с ее типом в момент …

А) присваивания значения;

Б) объявления переменной;

В) обращения к переменной

2. Возможность языка Python запросить структуру и тип объекта во время выполнения программы – это …

А) компиляция;

Б) интроспекция;

В) интерпретация

3. Для определения на какой тип данных ссылается переменная в языке Python, используется функция …

А) function (имя переменной);

Б) type (имя переменной);

В) range (имя переменной)

4. Выберете верное имя переменной

А) speedCar;

Б) 7 speed Car;

В) speed car

5. Выберете верное имя переменной

А) else;

Б) while;

В) volume

6. Python поддерживает _____ модель данных

А) ссылочную;

Б) объектную;

В) абстрактную

7. В языке Python после присваивания значения в переменной сохраняется …

А) ссылка на объект;
Б) тип переменной;
В) имя переменной

8. Выберете верный вариант позиционного присваивания переменных

А) >>>x, y, *z = (1, 2,3,4);
Б) >>>x, y, z = (1, 2,3,4);
В) >>>x, *y, *z = (1, 2,3,4);

9. Каким будет результат выполнения программного кода

```
>>> x=y=[5,8]
>>> y[1]=100
>>>x,y
```

А) ([5,8], [100,8]);
Б) ([5,100], [5,100]);
В) ([5,8], [5,100])

10. Каким будет результат выполнения программного кода

```
>>>x, y, *z, t = (1, 2, 3, 4, 5, 6, 7)
>>>x,y,z,t
```

А) (1,2,3,4,5,[6],7);
Б) ([1,2], [3,4],5, [6,7]);
В) (1,2,[3,4,5,6],7)

Выберите верный результат выполнения операции

```
>>> "8"*4
```

А) 32;
Б) '8888';
В) '44444444'

11. Выберите верный результат выполнения операции

```
>>> 8*4
```

А) 32;
Б) '8888';
В) '44444444'

12. Выберите верный результат выполнения операции

```
>>> [7,9,3]*2
```

А) [14,18,9];
Б) [49,81,9];
В) [7,9,3,7,9,3]

13. В языке Python операция слияния (склеивания) строк называется …

А) конкатенация;
Б) итерация;
В) повторение

14. Модуль, используемый в Python для создания программы с GUI-интерфейсом:

a) tkGUI;
b) tkinter;
c) PyGUI

15. Элемент графического интерфейса, представляющий собой область для ввода однострочного текста:

a) Label;
b) Text;
c) Entry

16. Контейнер, содержащий один или несколько элементов графического интерфейса:

a) Group;
b) Frame;

c) Entry

17. Программы с GUI-интерфейсом относятся к …
    a) событийно-ориентированным;
    b) процедурно-ориентированным;
    c) консольным

18. Кроссплатформенная, событийно-ориентированная графическая библиотека для разработки приложений с графическим интерфейсом, являющаяся частью стандартной поставки Python:
    a) PyQt;
    b) tkinter;
    c) PySide

19. Метод, предназначенный для размещения элемента интерфейса в заданной позиции и делающий его видимым при выводе на экран главного окна приложения:
    a) pack();
    b) position();
    c) show()

20. Для извлечения данных из элемента интерфейса Entry используется метод:
    a) data();
    b) reed_entry();
    c) get()

Вопросы для самостоятельного изучения:

1. Описание и формулирование задач процессов приобретения, поставки, разработки, эксплуатации и сопровождения ПО
2. Описание и формулирование задач процессов документирования, управления конфигурацией, обеспечения качества, верификации, совместной оценки, аудита, разрешения проблем и аттестации ПО
3. Понятие качества программного средства.
4. Документация, создаваемая и используемая в процессе разработки программных средств
5. Документы управления разработкой ПО
6. Документы, входящие в состав ПО
7. Определение качества ПО в стандарте ISO 9126. Многоуровневая модель качества ПО. Оценочные характеристики качества программного продукта
8. Стандарт ISO/IEC 12207-95: основные определения – система, модель жизненного цикла, квалификационные требования. Основные процессы, их содержание, работы и задачи процесса разработки
9. Прогностические модели процесса разработки: каскадная, RAD, спиральная
10. Объектный подход к разработке программных средств. Объектная модель
11. Метод восходящей и нисходящей разработки структуры программы
12. Отладка программного средства. Принципы и виды отладки
13. Контроль процесса разработки. Размерно- и функционально-ориентированные метрики
14. Сравнительный анализ структурного и объектно-ориентированного подходов к разработке ПО.

5. Оценочные материалы промежуточной аттестации по дисциплине

5.1. Экзамен (зачет) проводится с применением следующих методов (средств):

Экзамен принимается в устной форме, по билетам. Каждый экзаменационный билет включает два теоретических вопроса и практическое задание.

5.2. Оценочные материалы промежуточной аттестации

Компонент компетенции	Ключевой индикатор оценивания	Критерии оценивания
ОПК-6.2	Способен разрабатывать программы, пригодные для практического применения в области информационных систем и технологий	Демонстрирует знание методов и этапов реализации технологии программирования и умение разрабатывать программы с использованием структурного, функционального и объектно-ориентированного программирования

Типовые оценочные материалы промежуточной аттестации

Перечень вопросов для проведения экзамена

1. Общие принципы, основные подходы и этапы разработки программного обеспечения.
2. Модели жизненного цикла программного обеспечения.
3. Стандарты в области разработки и реализации программного обеспечения.
4. Методологические принципы разработки и реализации алгоритмов и программ.
5. Эволюция и перспективы развития технологий программирования.
6. Этапы эволюционного развития технологий программирования: неструктцрированное программирование, процедурное и модульное программирование, концепция объектно-ориентированного программирования, декларативное программирование.
7. Компонентный подход в технологиях программирования.
8. Обзор современных технологий программирования и тенденции их развития.
9. Концепция данных и основные конструкции языка программирования Python.
10. Программирование линейных, разветвляющихся, циклических алгоритмов.
11. Технологии работы со списками, кортежами, множествами, словарями
12. Технологии работы со строками.
13. Вложенные последовательности в Python.
14. Модули и пакеты Python. Технологии подключения и работы с модулями
15. Работа с датой и временем в Python.
16. Технологии работы с файлами
17. Событийно-ориентированное программирование.
18. Технология визуального проектирования и событийно-ориентированного программирования.
19. Этапы разработки GUI-приложения в модуле tkinter.
20. Основные виджеты модуля tkinter и технологии работы с ними.
21. Менеджеры размещения виджетов.
22. Понятие и виды функций в python. Встроенные функции.
23. Технологии создания пользовательских функций и модулей.
24. Методология (парадигма) объектно-ориентированного программирова**ния.** Основные принципы объектно-ориентированного программирования.
25. Атрибуты объектов и классов. Методы экземпляра класса.
26. Технологии работы с декораторами.
27. Наследование и полиморфизм.
28. Обзор библиотек Python для анализа данных.
29. Технологии анализа данных с использованием библиотеки Pandas.
30. Структура проекта в Django. Модели, представление, шаблоны.
31. Создание формы заявки и ее улучшение.
32. Фильтрация и поиск.
33. Публикация сайта.

Варианты практических (ситуационных) заданий для проведения экзамена

Задание 1

Создайте класс AvgList для получения среднего арифметического и медианы списка.

AvgList должен наследоваться от класса list, который является основным для списков в Python.

Класс должен содержать следующие методы:

mean() — возвращает среднее арифметическое.

median() — возвращает медиану или среднее значение из списка. Если количество элементов в списке четное, то медиана возвращает среднее между двумя центральными.

median_low() — возвращает медиану, но если количество элементов в списке четное, то возвращает меньшее (левое) из двух центральных элементов.

median_high() — возвращает медиану, но если количество элементов в списке четное, то возвращает большее (правое) из двух центральных элементов.

Задание 2

В файле matrix.txt содержится матрица M x N. Напишите программу transpose.py, которая транспонирует матрицу и сохраняет результат в файл transpose_matrix.txt.

Задание 3

Написать функцию **near_shops**, которая находит все магазины в радиусе R от заданной точки. Функция принимает три аргумента:

Координаты точки, вокруг которой реализуется поиск. В виде кортежа из двух элементов (X, Y).

Радиус поиска — R.

Список магазинов из кортежей вида ("название магазина", (Координата_X, Координата_Y))

Функция должна возвращать список из кортежей вида ("название магазина", Расстояние_до_магазина). Кортежи в списке должны быть отсортированы по расстоянию до магазина (от близких к дальним). В список должны попасть только магазины, которые расположены в пределах R.

Шкала оценивания

Оценка	**Критерии оценивания**
Отлично / зачтено	Оценка «отлично» («зачтено») выставляется обучающемуся, если в процессе промежуточной аттестации он демонстрирует глубокое знание теоретического программного материала и безошибочно применяет его при решении практических заданий
Хорошо / зачтено	Оценка «хорошо» («зачтено») выставляется обучающемуся, если в процессе промежуточной аттестации он демонстрирует хорошую степень освоения теоретического программного материала, применяет теоретические положения при решении практических заданий, допуская при этом отдельные незначительные ошибки
Удовлетворительно / зачтено	Оценка «удовлетворительно» («зачтено») выставляется обучающемуся, если он демонстрирует знания только основного программного материала, допускает неточности в формулировках, испытывает затруднения при выполнении практических заданий
Неудовлетворительно / не зачтено	Оценка «неудовлетворительно» («не зачтено») выставляется в случае, если обучающийся не освоил обязательный минимум знаний, умений и навыков в рамках дисциплины и допускает грубые ошибки при выполнении заданий промежуточной аттестации

6. Методические указания для освоения дисциплины

Обучение по дисциплине предполагает изучение курса на аудиторных (лекционных, практических) занятиях, самостоятельную работу обучающихся и проведение кон-

трольных мероприятий (текущей и промежуточной аттестации).

В начале учебного курса рекомендуется ознакомиться с рабочей программой дисциплины, обратив внимание на следующие ключевые моменты:

– перечень знаний, умений и навыков, которыми обучающийся должен овладеть в результате освоения дисциплины;

– тематический план проведения занятий;

– формы и материалы для проведения текущего контроля;

– учебники и учебные пособия по дисциплине, а также электронными ресурсы;

– оценочные материалы для проведения промежуточной аттестации.

С целью обеспечения успешного освоения курса обучающийся должен готовиться к лекционным и практическим занятиям.

На лекционных занятиях рекомендуется конспектировать основные положения излагаемого преподавателем материала. При подготовке к занятиям целесообразно самостоятельно прорабатывать лекционный материал, особое внимание следует уделять возникающим при этом вопросам, выделяя или выписывая указанные моменты для дальнейшего уточнения и обсуждения на последующих занятиях. В случае необходимости обращаться к преподавателю за консультацией. Таким образом, подготовка к лекции заключается в:

– внимательном прочтении материала предыдущей лекции;

– знакомстве с темой предстоящей лекции (по тематическому плану или информации лектора);

– предварительном знакомстве с учебным материалом по учебнику и учебным пособиям;

– уяснении места изучаемой темы в своей профессиональной подготовке;

– записи возможных вопросов преподавателю.

Практические занятия предполагают их проведение в различных формах (в том числе с использованием информационно-коммуникационных технологий) с целью систематизации, углубления и конкретизации полученных теоретических знаний, а также формирования навыков их применения при решении практических задач. Подготовка к практическим занятиям заключается в:

– внимательном прочтении теоретического материала лекций, относящихся к данному занятию, изучении учебного материала в рекомендуемых учебно-методических пособиях;

– ответе на приведенные в рабочей программе дисциплины вопросы для самостоятельной подготовки к занятиям лекционного, практического (семинарского) типов, лабораторным работам;

– выполнении заданий для самостоятельной работы по соответствующей теме.

В рамках практических занятий предусмотрено проведение мероприятий текущего контроля с целью выявления уровня освоения полученных знаний, умений, навыков и компетенций. Для осуществления текущего контроля в рабочей программе предусмотрены различные формы. При подготовке к контрольным мероприятиям рекомендуется ознакомиться с представленными в рабочей программе типовыми материалами для проведения текущего контроля успеваемости обучающихся.

Изучение дисциплины предполагает самостоятельное освоение обучающимся части учебного материала в соответствии с перечнем вопросов для самостоятельного изучения.

Подготовка к промежуточной аттестации. предполагает целенаправленную, регулярную, систематическую работу при изучении дисциплины в рамках аудиторных занятий и самостоятельной работы. Систематическое выполнение учебной работы на лекциях и практических занятиях, а также в рамках самостоятельной работы позволит успешно освоить дисциплину и создать хорошую базу для успешного прохождения промежуточной

аттестации. Непосредственно подготовку к процедуре аттестации рекомендуется осуществлять в соответствии с приведенным в рабочей программе перечнем типовых оценочных материалов для проведения промежуточной аттестации.

7. Учебная литература и ресурсы информационно-телекоммуникационной сети Интернет

7.1. Основная литература

1. Гниденко И. Г. Технологии и методы программирования: учебное пособие для вузов / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — Москва: Издательство Юрайт, 2022. — 235 с. — (Высшее образование). — ISBN 978-5-534-02816-4. — URL: https://urait.ru/bcode/489920
2. Федоров Д.Ю. Программирование на языке высокого уровня Python: учебное пособие для вузов / Д. Ю. Федоров. — 4-е изд., перераб. и доп. — Москва: Издательство Юрайт, 2022. — 214 с. — (Высшее образование). — ISBN 978-5-534-15733-8. — URL: https://urait.ru/bcode/509562

7.2. Дополнительная литература

1. Зубкова Т.М. Технология разработки программного обеспечения: учебное пособие / Т. М. Зубкова. — Санкт-Петербург: Лань, 2022. — 324 с. — ISBN 978-5-8114-3842-6. — Текст: электронный // Лань: электронно-библиотечная система. — URL: https://e.lanbook.com/book/206882. — Режим доступа: для авториз. пользователей.
2. Зыков С.В. Программирование. Объектно-ориентированный подход: учебник и практикум для вузов / С. В. Зыков. — Москва: Издательство Юрайт, 2022. — 155 с. — (Высшее образование). — ISBN 978-5-534-00850-0. — URL: https://urait.ru/bcode/490423
3. Зыков, С. В. Программирование. Функциональный подход: учебник и практикум для вузов / С. В. Зыков. — Москва: Издательство Юрайт, 2022. — 164 с. — (Высшее образование). — ISBN 978-5-534-00844-9. — URL: https://urait.ru/bcode/490870
4. Чернышев С.А. Основы программирования на Python: учебное пособие для вузов / С. А. Чернышев. — Москва: Издательство Юрайт, 2022. — 286 с. — (Высшее образование). — ISBN 978-5-534-14350-8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/496893 (дата обращения: 30.09.2022).

7.3. Нормативные правовые документы.

1. Национальная программа «Цифровая экономика Российской Федерации» (утв. протоколом заседания президиума Совета при Президенте Российской Федерации по стратегическому развитию и национальным проектам от 4 июня 2019 г. № 7). – URL: https://digital.gov.ru/ru/activity/directions/858/?utm_referrer =https%3a%2f%2fwww.yandex.ru%2f
2. Указ Президента Российской Федерации от 09.05.2017 г. № 203 «О Стратегии развития информационного общества в Российской Федерации на 2017-2030 годы» – URL: http://www.kremlin.ru/acts/bank/41919

7.4. Интернет-ресурсы

1. Официальный портал СИУ РАНХиГС – URL: http://orel.ranepa.ru.
2. Образовательная платформа Юрайт – URL: https://urait.ru/
3. ЭБС Лань – URL: https://e.lanbook.com/
4. ЭБС Znanium – URL: https://znanium.com/
5. Официальный сайт компании «Консультант Плюс» [Электронный ресурс]. – Режим доступа: http://www.consultant.ru/, свободный.
6. Информационно-правовой портал [Электронный ресурс]. – Режим доступа: http://www.garant.ru, свободный.

7.5. Иные источники.

Не предусмотрены

8. Материально-техническая база, информационные технологии, программное обеспечение и информационные справочные системы

Материально-техническая база

Наименование специальных помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы
Учебная аудитория для проведения занятий лекционного типа, семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	Учебная мебель, проекционное оборудование, доска,
Учебная аудитория для проведения занятий семинарского типа, лабораторных занятий, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	Учебная мебель, проекционное оборудование, доска.
Компьютерный класс (аудитория для выполнения самостоятельной работы) /Учебная аудитория для проведения занятий семинарского типа, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	Учебная мебель, проекционное оборудование, компьютеры, доска. Компьютеры с выходом в сеть «Интернет» и ЭИОС СИУ РАНХиГС
Помещения для хранения и профилактического обслуживания оборудования	Комплекты специализированной мебели для хранения оборудования

Программное обеспечение

Наименование программного обеспечения	Назначение и тип лицензии программного обеспечения
7-Zip	Архиватор с высокой степенью сжатия Свободная лицензия
IDLE Python	Интегрированная среда разработки и обучения на языке Python Свободная лицензия
Adobe reader – Russian	Просмотр, печать и внесение аннотаций в документы PDF. Свободная лицензия
Adobe Acrobat Reader DC– Russian	Чтение, печать и рецензирование файлов PDF. Свободная лицензия
Google Chrome	Браузер Свободная лицензия
Java	Представляет собой язык программирования и платформу вычислений Свободная лицензия
Microsoft Visual C++	Интегрированная среда разработки приложений на языке C++. Свободная лицензия
Mozilla FireFox	Браузер. Свободная лицензия
Opera	Браузер. Свободная лицензия
Yandex Браузер	Браузер Свободная лицензия

Office Professional Plus 2007 Office Professional Plus 2010 Office Standard 2007 Office Standard 2010 Office Standard 2013	Пакет офисных приложений. Лицензированная версия
Консультант Плюс	Справочно-правовая система