

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ПОВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ



УТВЕРЖДАЮ
Декан ФГО

УТВЕРЖДАЮ /А.В. Артамонова/
(Ф.И.О. декана (директора института))

14.02.2025 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

М.1.2.4 Программирование лингвистических задач на языке Python

(код и наименование дисциплины по учебному плану)

Направление подготовки
(специальность)

45.04.02 Лингвистика

Квалификация выпускника

Магистр

(бакалавр/магистр/специалист)

Программа магистратуры

Цифровая лингвистика

Курс 1, 2

Семестр 2, 3

Распределение учебного времени

Трудоемкость по учебному плану	216 / 6	часов/зачетных единиц
Лекции	30	часов
Лабораторные работы	-	часов
Практические занятия	30	часов
Иная контактная работа	-	часов
Всего контактной работы (без учета экз.)	60	часов
Контактная работа по экзамену	-	часов
Курсовой проект (работа)	3	семестр
Самостоятельная работа обучающихся (без учета экз.)	156	часов
Самостоятельная работа по подготовке к экзамену	-	часов
Экзамен	-	семестр
Зачет	2	семестр
БРК, ДЗ	3	семестр

(год)

Программа составлена в соответствии с требованиями ФГОС ВО направления подготовки (специальности) 45.04.02 Лингвистика

Программу составили:

доцент	ИЯиЛ	СОГЛАСОВАНО	К.О. Иванов
(должность)	(кафедра)		(И.О. Фамилия)

РАССМОТРЕНА и ОДОБРЕНА на заседании кафедры, за которой закреплена дисциплина
Кафедра иностранных языков и лингвистики

(наименование кафедры)		
29.01.2025	протокол №	4
(дата)		
Заведующий кафедрой	СОГЛАСОВАНО	О.В. Филипчук
		(И.О. Фамилия)

Рабочая программа СОГЛАСОВАНА с факультетом (институтом), выпускающей(ими) кафедрой(ами).

СООТВЕТСТВУЕТ действующей ОП.

Заведующий кафедрой	СОГЛАСОВАНО	О.В. Филипчук
		(И.О. Фамилия)

Председатель методической комиссии факультета (института), в который входит выпускающая кафедра

СОГЛАСОВАНО	А.В. Артамонова
	(И.О. Фамилия)

Эксперт(ы): Лазарева Е.А. , директор Школы изучения иностранных языков «Лингва»

Рабочая программа проверена и зарегистрирована в УМЦ 17.02.2025 г.

Специалист учебно-методического центра СОГЛАСОВАНО /Т.А. Смирнова/

Раздел 1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Целью освоения дисциплины является достижение планируемых результатов обучения, соответствующих установленным в ОПОП индикаторам достижения компетенций:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Результаты обучения
1. УК-5 Способен анализировать и учитывать разнообразие культур в процессе межкультурного взаимодействия	ИД-1 ук-5 Анализирует современное состояние поликультурного общества на основе знаний межкультурной коммуникации и базы лингвистических данных (лингвистические корпуса, словари, тезаурусы)	знания: Знание культурных различий и сходств в общении. умения: Умение проводить критический анализ текстов, связанных с поликультурным обществом. навыки: Навыки работы с лингвистическими корпусами, словарями и тезаурусами.
	ИД-2 ук-5 Демонстрирует понимание национально-культурных различий в процессе межкультурного общения и взаимодействия как в режиме реального времени, так и	знания: Знание нескольких языков или, по крайней мере, базовое владение основами языка собеседника. умения: Умение распознавать и учитывать культурные различия в речи, жестах, выражениях и общении в целом. навыки: Навыки общения как в устной, так и в письменной форме.
2. ПК-4 Способен осуществлять ведение процедуры медиации и разрабатывать методику выполнения аналитических работ	ИД-1 ПК-4 использует корпуса языков и инструменты их обработки, системы машинного перевода, технологию искусственного интеллекта для анализа текстовой информации	знания: Знание основных языковых явлений и правил, в том числе морфологии и синтаксиса. умения: Умение использовать инструменты машинного обучения для обработки и анализа текстов. навыки: Навыки программирования для работы с инструментами обработки текста и машинного обучения.
	ИД-2 ПК-4 демонстрирует владение технологиями устного и письменного перевода с использованием систем машинного перевода и обработки естественных языков	знания: Знание тематических областей, таких как медицина, юриспруденция, техника и т.д., в которых человек работает. умения: Умение работать с платформами и инструментами для обработки естественных языков (Natural Language Processing, NLP). навыки: Навыки преодоления ограничений систем машинного перевода и улучшения качества переводов.

	ИД-3 пк-4 использует инструменты ИТ, лингвистические корпуса и базы данных для выявления, анализа и описания мировых практик и проведения аналитических работ	знания: Знание принципов создания, структуры и использования лингвистических корпусов. умения: Умение собирать и синтезировать информацию из различных источников. навыки: Навыки использования инструментов NLP для извлечения информации и выделения ключевых элементов из текстов.
--	---	--

Раздел 2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина относится к части, формируемой участниками образовательных отношений ОПОП.

Дисциплина является обязательной

Для продолжения формирования заявленных компетенций необходимы знания предшествующих дисциплин: Практическая грамматика (УК-5), Web-дизайн и Web-жанры (УК-5), Язык международных конференций (УК-5)

Изучаемая дисциплина является основой для продолжения формирования указанных компетенций в следующих дисциплинах: Конфликтология (ПК-4), Язык презентаций (УК-5), Практический курс первого иностранного языка (УК-5), Методы искусственного интеллекта в компьютерной лингвистике (ПК-4), Практикум по культуре общения на иностранном языке (УК-5); практиках: Преддипломная практика (УК-5); государственной итоговой аттестации в форме: Подготовка к процедуре защиты и защита выпускной квалификационной работы (УК-5), Подготовка к процедуре защиты и защита выпускной квалификационной работы (ПК-4)

Раздел 3. ОПИСАНИЕ ОБРАЗОВАТЕЛЬНЫХ ТЕХНОЛОГИЙ

Для формирования заявленных компетенций используются методологические технологии, реализующие деятельностный, личностно-ориентированный, практико-ориентированный подходы.

Основными стратегическими технологиями являются: практические занятия, лекционные занятия, процедуры самообучения

На достижение конкретных целей обучения направлены применяемые тактические технологии: классическая лекция, лекция с элементами мозгового штурма, мини-проекты, проблемная лекция

Раздел 4. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

2 семестр

Виды и темы занятий	Количество часов	Формируемые компетенции
Основы программирования на Python	108	ПК-4, УК-5
Лекция. Введение в Python: Основы синтаксиса и структура программы: История Python и его философия. Основные элементы синтаксиса: переменные, операторы, выражения. Блоки кода и управляющие конструкции.	2	
Лекция. Типы данных и переменные в Python:	2	

Числовые типы данных: целые числа, числа с плавающей точкой. Строки и операции над ними. Списки, кортежи, множества и словари.	
Лекция. Управление потоком выполнения программы: Условные операторы: if, elif, else. Циклы: for и while. Прерывание циклов: break и continue.	2
Лекция. Функции в Python: Определение функций и передача аргументов. Возвращаемые значения и область видимости переменных. Рекурсивные функции.	2
Лекция. Работа с файлами и обработка исключений: Открытие, чтение и запись файлов. Обработка исключений: try, except, finally. Работа с контекстными менеджерами.	2
Лекция. Объектно-ориентированное программирование (ООП) в Python: Основные принципы ООП: инкапсуляция, наследование, полиморфизм. Классы и объекты в Python. Специальные методы (магические методы).	2
Лекция. Работа с модулями и пакетами Создание и использование модулей. Импортирование и использование сторонних библиотек. Создание и организация пакетов.	2
Лекция. Работа с базами данных в Python: Введение в работу с базами данных SQLite. Использование библиотек для работы с другими базами данных (например, SQLAlchemy).	2
Практическое занятие. Введение в Python: Установка Python и настройка среды разработки. Основы синтаксиса: переменные, типы данных, операторы. Ввод/вывод данных.	2
Практическое занятие. Управляющие конструкции: Условные операторы (if, elif, else). Циклы (for, while). Применение циклов и условий для решения задач.	2
Практическое занятие. Функции и модули: Определение функций. Параметры функций и возвращаемые значения. Импорт и использование модулей.	2
Практическое занятие. Структуры данных: Списки и операции с ними. Кортежи. Словари и множества.	2
Практическое занятие. Работа с файлами: Чтение и запись в текстовые файлы. Работа с форматами данных (JSON, CSV).	2
Практическое занятие. Обработка ошибок: Исключения и их обработка. Защита кода от ошибок.	2

Практическое занятие. ООП в Python: Создание классов и объектов. Наследование и полиморфизм.	2	
Практическое занятие. Работа с API: Запросы к внешним API. Обработка JSON-ответов.	2	
Задания для самостоятельной работы, в том числе выполнение самостоятельное изучение разделов курса: 1. Веб-разработка на Python; Основы работы с веб-фреймворками (например, Flask, Django). Обработка HTTP-запросов и создание веб-приложений. 2. Тестирование и отладка в Python: Написание модульных тестов. Отладка кода с использованием инструментов, таких как pdb. 3. Параллельное и асинхронное программирование в Python: Многозадачность и потоки выполнения. Асинхронное программирование с использованием async/await. 4. Введение в научные вычисления с использованием библиотеки NumPy: Создание и работа с массивами. Математические операции и операции индексации.	76	
Иная контактная работа:	0	

3 семестр

Виды и темы занятий	Количество часов	Формируемые компетенции
Программирование лингвистических задач на языке	108	ПК-4, УК-5
Лекция. Введение в лингвистическое программирование: Обзор задач, связанных с обработкой естественного языка (Natural Language Processing, NLP). Важность лингвистического анализа в современном программировании.	2	
Лекция. Основы работы с текстом в Python: Чтение и запись текстовых файлов. Основы работы со строками и регулярными выражениями.	2	
Лекция. Токенизация и сегментация текста: Разделение текста на токены. Сегментация предложений.	2	
Лекция. Частеречная разметка: Определение частей речи в тексте. Использование библиотек для частеречной разметки в Python.	2	
Лекция. Лемматизация и стемминг: Приведение слов к базовой форме. Работа с библиотеками для лемматизации и стемминга.	2	
Лекция. Извлечение ключевых слов и фраз: Методы автоматического извлечения ключевых элементов из текста.	2	
Лекция. Анализ сентимента: Определение тональности текста. Использование методов машинного обучения для анализа сентимента.	2	
Практическое занятие. Анализ и обработка текста: реализация функции для токенизации текста на слова и сегментации на	2	

предложения.		
Практическое занятие. Токенизация и сегментация текста: Реализация функции для токенизации текста на слова и сегментации на предложения.	2	
Практическое занятие. Частеречная разметка: реализация программы для определения частей речи в предложении	2	
Практическое занятие. Лемматизация и стемминг: Создание программы, использующую библиотеки для лемматизации и стемминга слов в тексте.	2	
Практическое занятие. Извлечение ключевых слов и фраз: Разработка функции для автоматического извлечения ключевых слов из текста.	2	
Практическое занятие. Сентиментальный анализ текстов	2	
Практическое занятие. Машинное обучение в лингвистическом программировании:	2	
Задания для самостоятельной работы, в том числе выполнение курсового проекта/работы		
подготовка к практическим занятиям, подготовка курсовой работы	80	
выполнение курсового проекта/работы	0	
Иная контактная работа:	0	

Раздел 5. МЕТОДИЧЕСКИЕ УКАЗАНИЯ ДЛЯ ОБУЧАЮЩИХСЯ ПО ОСВОЕНИЮ ДИСЦИПЛИНЫ

Изучение дисциплины рекомендуется начать с ознакомления с рабочей программой, ее структурой и содержанием разделов. Учебный материал структурирован, изучение дисциплины осуществляется в тематической последовательности.

Занятия лекционного типа дают систематизированные знания по дисциплине, концентрируют внимание на наиболее сложных и важных вопросах. Во время лекционных занятий рекомендуется вести конспектирование учебного материала; обращать внимание на формулировки и категории, раскрывающие суть проблемы, явления или процесса; зафиксировать выводы и практические рекомендации. Подготовка к **занятиям семинарского типа** включает ознакомление с планом практического занятия; работу с конспектом лекций, выполнение домашнего задания, работу с учебной и учебно-методической литературой, научными изданиями и электронными образовательными ресурсами, рекомендованными рабочей программой дисциплины.

Содержание **самостоятельной работы** определяется рабочей программой дисциплины, оценочными и методическими материалами, заданиями и указаниями преподавателя. Самостоятельная работа может осуществляться в аудиторной и внеаудиторной формах. Эффективным средством осуществления самостоятельной работы является электронная информационно-образовательная среда университета, которая обеспечивает доступ к образовательной программе, рабочей программе дисциплины, к электронным библиотечным системам, профессиональным базам данных и информационным справочным системам.

Изучение дисциплины включает выполнение курсовой работы. Курсовая работа по дисциплине "Программирование лингвистических задач на языке Python" может иметь различные задачи в зависимости от темы, оформляется в соответствии с ГОСТ 7.32–2017.

Рекомендуется, чтобы курсовая работа содержала следующие разделы:

1.

Введение:

? Формулировка актуальности выбранной темы.

? Цель и задачи исследования.

? Обоснование выбора языка программирования Python для решения лингвистических

2. Обзор литературы:
 - ? Анализ существующих методов и подходов к решению лингвистических задач на языке программирования Python.
 - ? Описание инструментов и библиотек, широко используемых в лингвистическом программировании.
3. Постановка задачи:
 - ? Формулировка конкретной задачи, которую предстоит решить в рамках курсовой работы.
 - ? Определение критериев успешного решения задачи.
4. Анализ требований:
 - ? Изучение и формулировка требований к входным и выходным данным.
 - ? Разработка алгоритмов, которые будут использоваться для решения задачи.
5. Программная реализация:
 - ? Написание программного кода на языке Python для решения поставленной задачи.
 - ? Комментирование кода для облегчения понимания.
6. Результаты и обсуждение:
 - ? Представление полученных результатов в форме графиков, таблиц, диаграмм.
 - ? Обсуждение достигнутых результатов и сравнение с существующими методами.
7. Заключение:
 - ? Подведение итогов исследования.
 - ? Выводы по результатам работы.
 - ? Рекомендации по дальнейшим исследованиям в данной области.
8. Список литературы:
 - ? Формирование списка использованных источников согласно выбранному стандарту (например, APA, MLA).
9. Приложения:
 - При необходимости включение дополнительных материалов, таких как исходный код, дополнительные графики, результаты тестирования и пр.

- Темы курсовых работ:
1. Автоматическая обработка естественного языка (Natural Language Processing, NLP):
 - ? Разработка программы для извлечения ключевых слов из текста на русском/английском языке.
 - ? Создание алгоритма для определения тональности текста (анализ сентимента).
 2. Морфологический анализ текста:
 - ? Реализация морфологического анализатора для русского языка на основе библиотек Python.
 - ? Создание программы для лемматизации текста с использованием морфологического анализа.
 3. Машинный перевод:
 - ? Разработка простого системного переводчика с одного языка на другой с использованием машинного обучения.
 - ? Исследование и сравнение существующих библиотек и методов машинного перевода на Python.
 4. Анализ частоты слов:
 - ? Создание программы для анализа частоты употребления слов в тексте на определенном языке.
 - ? Сравнение результатов анализа для различных литературных произведений.
 5. Генерация текста:
 - ? Реализация программы для генерации текста на основе заданных правил и структур.
 - ? Применение методов генерации текста с использованием искусственных нейронных сетей.

6. Распознавание речи:
- ? Разработка системы распознавания и интерпретации речи на Python.
 - ? Сравнение точности распознавания для различных языков и диалектов.
7. Классификация текстов:
- ? Реализация программы для классификации текстов по темам с использованием машинного обучения.
 - ? Обзор методов и библиотек для решения задачи классификации.
8. Исследование синтаксической структуры текста:
- ? Разработка алгоритма для анализа синтаксической структуры предложений в тексте.
 - ? Исследование влияния структуры на понимание смысла текста.
9. Создание чат-бота с естественным языком общения:
- ? Реализация чат-бота, способного понимать и генерировать текст с использованием NLP.
10. Анализ и сравнение языков программирования для лингвистических задач:
- ? Сравнение эффективности различных языков программирования при решении задач в области обработки естественного языка.

Периодичность проведения, формы текущего контроля успеваемости, система оценивания хода освоения дисциплин представлены в рабочей программе. Формой промежуточной аттестации по дисциплине является зачёт, балльно-рейтинговый контроль.

Раздел 6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ И УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

6.1. Учебно-методическое обеспечение

№№ п/п	Список используемой литературы	Количество экземпляров печатных изданий, имеющих в библиотеке, или электронный адрес издания (ресурса) в сети Интернет
УЧЕБНЫЕ, УЧЕБНО-МЕТОДИЧЕСКИЕ И НАУЧНЫЕ ИЗДАНИЯ		
1.	Архипов, О. Г. Программирование. Сборник задач [Электронный ресурс] : учебное пособие / Архипов О. Г., Батасова В. С., Гречкина П. В., Зубов В. С., Воробьева И. А., Ионова Т. В., Костина М. Б., Крюков А. А., Чибизова Н. В., Щербин В. М., Марана М. М. Санкт-Петербург: Лань, 2022. - 140 с. ISBN 978-5-8114-3857-0.	https://e.lanbook.com/book/223418
2.	Щерба, А. В. Программирование на Python. Первые шаги [Электронный ресурс] / Щерба А. В. Москва: Лаборатория знаний, 2022. - 250 с. ISBN 978-5-93208-578-3.	https://e.lanbook.com/book/221678
3.	Федоров, Дмитрий Юрьевич. Программирование на языке высокого уровня Python [Текст : Электронный ресурс] : учебное пособие для вузов / Д. Ю. Федоров. 5-е изд. Москва: Юрайт, 2023. - 210 с ISBN 978-5-534-14638-7.	https://urait.ru/book/programirovanie-na-yazyke-vysokogo-urovnya-python-532868
4.	Казанский, Александр Анатольевич. Программирование на Visual C# [Текст : Электронный ресурс] : учебное пособие для вузов / А. А. Казанский. 2-е изд. Москва: Юрайт, 2022. - 192 с ISBN 978-5-534-12338-8.	https://urait.ru/bcode/470261
5.	Рацеев, С. М. Программирование. Лабораторный	

	практикум [Электронный ресурс] : учебное пособие для вузов / Рацеев С. М. Санкт-Петербург: Лань, 2023. - 104 с. ISBN 978-5-507-45194-4.	https://e.lanbook.com/book/292907
6.	Павловская, Т. А. Программирование на языке C++ [Электронный ресурс] / Павловская Т. А. 2-е изд. Москва: ИНТУИТ, 2016. - 154 с.	https://e.lanbook.com/book/100409
7.	Кудрина, Е. В. Программирование на языке C#: разработка консольных приложений [Электронный ресурс] / Кудрина Е. В., Огнева М. В., Портенко М. С. 2-е изд. Москва: ИНТУИТ, 2016. - 366 с.	https://e.lanbook.com/book/100412
8.	Павловская, Т. А. Программирование на языке высокого уровня C# [Электронный ресурс] / Павловская Т. А. 2-е изд. Москва: ИНТУИТ, 2016. - 245 с.	https://e.lanbook.com/book/100413
9.	Алексеев, Е. Р. Программирование на языке C++ в среде Qt Creator [Электронный ресурс] / Алексеев Е. Р., Злобин Г. Г., Костюк Д. А., Чеснокова О. В.; Чмыхало А.С. 2-е изд. Москва: ИНТУИТ, 2016. - 715 с.	https://e.lanbook.com/book/100414
10.	Грацианова, Т. Ю. Программирование в примерах и задачах [Электронный ресурс] / Грацианова Т. Ю. 6-е изд. Москва: Лаборатория знаний, 2020. - 373 с. ISBN 978-5-00101-927-5.	https://e.lanbook.com/book/152040
ЭЛЕКТРОННЫЕ ОБРАЗОВАТЕЛЬНЫЕ РЕСУРСЫ		
1.	Научная электронная библиотека eLIBRARY.RU	http://elibrary.ru
2.	Научная электронная библиотека «Киберленинка»	http://cyberleninka.ru
ПРОФЕССИОНАЛЬНЫЕ БАЗЫ ДАННЫХ И ИНФОРМАЦИОННЫЕ СПРАВОЧНЫЕ СИСТЕМЫ		
1.	Справочно-правовая система Консультант+	http://www.consultant.ru
2.	Информационно-правовой портал Гарант	http://www.garant.ru
3.	Профессиональные справочные системы Техэксперт	http://www.cntd.ru

6.2. Материально-техническая база и программное обеспечение

№№ п/п	Аудитории для проведения учебных занятий, самостоятельной работы и проведения государственной итоговой аттестации	Перечень основного оборудования	Программное обеспечение
1.	531 (III)	ПК B112,2 420W/Intel Celeron Dual-Core E3300/кл,мышь,фильт,мон. VA1931 (5)	Microsoft Windows Enterprise, Справочная правовая система "Консультант Плюс", Microsoft Office Standard, Агент Dr.Web, Комплект ГАРАНТ-Мастер, Microsoft Access, Microsoft Visio Professional, Microsoft Project Professional, Microsoft Visual Studio Enterprise, Комплект ПО для решения основных

Раздел 7. ФОРМЫ КОНТРОЛЯ ОСВОЕНИЯ ДИСЦИПЛИНЫ/ ФОНД ОЦЕНОЧНЫХ СРЕДСТВ

Критерии оценивания индикаторов достижения компетенций направлены на:

- усвоение теоретического материала (объем знаний, глубина усвоения), предусмотренного рабочей программой;
- умение излагать материал (четкость, грамотность изложения материала, точность и полнота воспроизведения учебного материала);
- умение применять теоретические знания при решении практических заданий.

Шкала оценивания представлена ниже.

Уровень сформированности элементов компетенции	Критерии оценивания	Шкала оценивания
Пороговый уровень	Обучающийся имеет знания основного материала, проявляет умение логично его излагать, но может допускать неточности в изложении материала, недостаточно правильные формулировки, испытывает затруднения в выполнении практических заданий.	удовлетворительно
Продвинутый уровень	Обучающийся твердо знает программный материал, излагает его грамотно и по существу, не допускает существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения	хорошо
Высокий уровень	Обучающийся глубоко и прочно усвоил программный материал, грамотно и логически стройно его излагает, дает исчерпывающие ответы на поставленные вопросы. В ответе тесно увязывается теория с практикой, при этом обучающийся не затрудняется с ответом при видоизменении задания, свободно справляется с задачами, вопросами и другими видами применения знаний, показывает знакомство с монографической литературой, периодическими изданиями, правильно обосновывает принятые решения, свободно владеет разносторонними навыками, приемами выполнения практических работ	отлично

7.1. Текущий контроль успеваемости

Текущий контроль успеваемости обеспечивает оценивание хода освоения дисциплины (модуля) и производится с применением технологии рейтингового контроля в соответствии с технологической картой дисциплины. Порядок составления технологической карты и алгоритм проведения процедуры оценивания видов деятельности обучающихся, направленных на освоение знаний, умений, навыков и/или опыта деятельности, по накопительной системе в баллах устанавливается положением о системе РИТМ в ФГБОУ ВО «ПГТУ»

7.2. Промежуточная аттестация обучающихся

Промежуточная аттестация обучающихся направлена на оценивание результатов обучения

по дисциплине (модулю) и проводится с использованием фондов оценочных средств.

Примеры типовых контрольных заданий из базы фонда оценочных средств по образовательной программе.

2 семестр:

1. Какой из следующих кодов правильно объявляет переменную с именем "count" и присваивает ей значение 10?

- a) `int count = 10`
- b) `count = 10`
- c) `count := 10`
- d) `set count = 10`

2. Что такое PEP 8 в контексте Python?

- a) Генератор случайных чисел.
- b) Соглашение о стиле кода.
- c) Библиотека для работы с временем.
- d) Встроенная функция для подсчета элементов в списке.

3. Какой результат будет у выражения `3 * 'abc'`?

- a) 9
- b) `abcbcabcb`
- c) `['abc', 'abc', 'abc']`
- d) Ошибка

4. Как создать список, содержащий квадраты чисел от 1 до 5?

- a) `[x*x for x in range(1, 6)]`
- b) `(x*x for x in range(1, 6))`
- c) `{x*x for x in range(1, 6)}`
- d) `list(x*x for x in range(1, 6))`

5. Какие из перечисленных операторов используются для выполнения условных операций в Python?

- a) `switch`
- b) `if, else`
- c) `select`
- d) `check`

6. Какие из следующих методов используются для работы со строками в Python?

- a) `substring()`
- b) `index()`
- c) `add()`
- d) `push()`

7. Как вывести элемент с индексом 2 из списка `my_list`?

- a) `print(my_list[2])`
- b) `print(my_list.index(2))`
- c) `print(my_list[1])`
- d) `print(my_list.get(2))`

8. Какой оператор используется для конкатенации (слияния) строк в Python?

- a) `&`
- b) `+`
- c) `*`
- d) `.`

9. Что делает метод `strip()` для строки?

- a) Удаляет все пробелы в начале и конце строки.
- b) Разделяет строку на подстроки.
- c) Заменяет символы в строке.
- d) Извлекает подстроку из строки.

10. Какие из перечисленных библиотек часто используются для обработки естественного языка (NLP) в Python?

- a) `matplotlib`
- b) `numpy`
- c) `pandas`
- d) `nlTK`

Правильные ответы:

- 1. b) `count = 10`
- 2. b) Соглашение о стиле кода.
- 3. b) `abcaabcabc`
- 4. a) `[x*x for x in range(1, 6)]`
- 5. b) `if, else`
- 6. b) `index()`
- 7. a) `print(my_list[2])`
- 8. b) `+`
- 9. a) Удаляет все пробелы в начале и конце строки.
- 10. d) `nlTK`

3 семестр:

1. Что представляет собой процесс токенизации в обработке естественного языка (NLP)?

- a) Преобразование текста в последовательность токенов.
- b) Замена ключевых слов в тексте.
- c) Объединение предложений в тексте.
- d) Удаление стоп-слов из текста.

2. Какая из следующих библиотек Python обеспечивает возможности частеречной разметки текста?

- a) Matplotlib
- b) NumPy
- c) spaCy
- d) Requests

3. Что такое лемматизация в контексте обработки естественного языка (NLP)?

- a) Удаление стоп-слов из текста.
- b) Приведение слов к их базовой форме.
- c) Извлечение ключевых фраз из текста.
- d) Группировка слов по частоте встречаемости.

4. Какой метод библиотеки NLTK можно использовать для извлечения ключевых слов из текста?

- a) `extract_keywords()`
- b) `keywords.extract()`
- c) `nltk.keyword_extraction()`
- d) `nltk.FreqDist()`

5. Какие библиотеки обычно используются для анализа сентимента в тексте с помощью машинного обучения?

- a) TensorFlow
- b) Keras
- c) NLTK
- d) scikit-learn

6. Какую задачу решает алгоритм стемминга в обработке естественного языка?

- a) Преобразование слова к его базовой форме.
- b) Извлечение ключевых фраз из текста.
- c) Разделение предложений на токены.
- d) Определение тональности текста.

7. Какие библиотеки Python предоставляют возможности для генерации текста на основе заданных данных?

- a) spaCy
- b) NLTK
- c) GPT-3
- d) scikit-learn

8. Как осуществить токенизацию предложений с использованием библиотеки spaCy в Python?

- a) `sent_tokenize()`
- b) `spaCy.sentence_tokenize()`
- c) `spaCy.tokenize_sentences()`
- d) `spaCy.sent_tokenizer()`

Правильные ответы:

1. а) Преобразование текста в последовательность токенов.
2. с) spaCy
3. б) Приведение слов к их базовой форме.
4. а) `extract_keywords()`
5. с) NLTK
6. а) Преобразование слова к его базовой форме.
7. с) GPT-3
8. а) `sent_tokenize()`

Перечень вопросов для проведения промежуточной аттестации

2 семестр:

1. Какова разница между Python 2 и Python 3?
2. Что такое PEP 8, и почему это важно для разработчиков на Python?
3. Как создать комментарий в Python?
4. Какие базовые типы данных поддерживает Python?
5. Что такое списки (lists) в Python и как их можно модифицировать?
6. Какие управляющие конструкции используются в Python?
7. В чем разница между for и while циклами?
8. Как использовать условные операторы if, elif и else?
9. Как объявить функцию в Python?
10. Что такое аргументы по умолчанию (default arguments) в функциях?
11. Каким образом можно импортировать модуль в Python?
12. Что такое кортежи (tuples) и в чем их отличие от списков?
13. Как создать словарь (dictionary) в Python и как обратиться к его элементам?
14. Что такое множества (sets) и для чего они используются?
15. Что такое исключение (exception) в Python?
16. Как обработать исключение с использованием блока try-except?
17. Как создать класс в Python?
18. В чем разница между методами и атрибутами класса?
19. Как реализовать наследование в Python?
20. Как открыть и прочитать содержимое файла в Python?
21. Как записать данные в файл в Python?

3 семестр:

1. Как можно прочитать текстовый файл и провести базовый лингвистический анализ с использованием Python?
2. Что такое токенизация и как ее реализовать в Python?
3. Как реализовать сегментацию предложений в тексте на уровне кода?
4. Как провести частеречную разметку (Part-of-Speech tagging) текста с использованием библиотеки NLTK или spaCy?
5. В чем заключается лемматизация и как ее можно выполнить с использованием библиотеки?
6. Как работает стемминг, и какую библиотеку можно использовать для стемминга слов?
7. Как создать программу для анализа сентимента текста с использованием машинного обучения?
8. Какие задачи обработки естественного языка могут быть решены с использованием методов машинного обучения?
9. Как использовать библиотеку scikit-learn для создания модели классификации текста?

10. Какие функции библиотеки NLTK могут быть использованы для анализа текста?
11. В чем отличие библиотеки spaCy от NLTK, и когда предпочтительно использовать одну из них?
12. Как создать программу для извлечения ключевых слов из текста?
13. Как использовать модели для генерации текста на основе заданного контекста?
14. Какие вызовы API можно использовать для обработки естественного языка в Python?
15. Каким образом можно обработать текстовые данные с использованием регулярных выражений в Python?
16. Какие проблемы могут возникнуть при обработке текста на естественном языке, и как их можно решить?