Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Максимов Алексей Борисович
Должность: директор департамента по образовательной политике
Дата подписания: 23.05.2026 08:27:25
Уникальный программный ключ:
8db180d1a3f02ac9e60521a5672742735885[illegible]

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)

Передовая инженерная школа технологического лидерства FDR

УТВЕРЖДАЮ
Директор
_____ /П.Итурралде /
«26» февраля 2026г.

АБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Методы обработки данных с использованием языка программирования Python

Направление подготовки
01.04.02. Прикладная математика и информатика

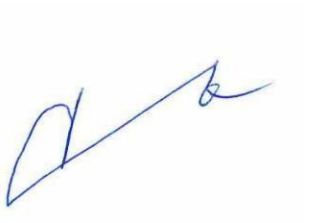
Профиль
Программная инженерия в автомобилестроении

Квалификация
магистр

Формы обучения
очная

Москва, 2026 г.

Разработчик(и):

Доцент, к.т.н.  /Ю.М.Фурлетов/

Согласовано:
Отдел организации
и управления учебным
процессом  /Д.Т.Хамдамова/

Руководитель
образовательной программы
преподаватель, к.т.н.,  /В.В.Петин/

Содержание

1. Цели, задачи и планируемые результаты обучения по дисциплине 4
2. Место дисциплины в структуре образовательной программы 5
3. Структура и содержание дисциплины 5
   - 3.1. Виды учебной работы и трудоемкость 5
   - 3.2. Тематический план изучения дисциплины 5
   - 3.3. Содержание дисциплины 6
   - 3.4. Тематика семинарских/практических и лабораторных занятий 8
   - 3.5. Тематика курсовых проектов (курсовых работ) 9
4. Учебно-методическое и информационное обеспечение 9
   - 4.1. Нормативные документы и ГОСТы 9
   - 4.2. Основная литература 9
   - 4.3. Дополнительная литература 10
   - 4.4. Электронные образовательные ресурсы 10
   - 4.5. Лицензионное и свободно распространяемое программное обеспечение 10
   - 4.6. Современные профессиональные базы данных и информационные справочные системы 10
5. Материально-техническое обеспечение 11
6. Методические рекомендации 11
   - 6.1. Методические рекомендации для преподавателя по организации обучения 11
   - 6.2. Методические указания для обучающихся по освоению дисциплины 12
7. Фонд оценочных средств 13
   - 7.1. Методы контроля и оценивания результатов обучения 14
   - 7.2. Шкала и критерии оценивания результатов обучения 14
   - 7.3. Оценочные средства 14

1. Цели, задачи и планируемые результаты обучения по дисциплине

Специалисту в области анализа данных в своей практической деятельности необходимо решать проблемы, компьютеры могут только выполнять программы обработки, реализующие определенные алгоритмы. Для использования компьютеров в решении практических проблем необходимо представить их в виде комплекса взаимосвязанных задач, выбрать методы решения этих задач и эффективные алгоритмы их реализации, разработать соответствующее программное обеспечение. В курсе рассматриваются содержание, трудности и особенности процесса анализа данных и разработки прикладных программных систем для их решения на языке Python.

Задачи дисциплины:

– ознакомиться с базовыми понятиями и элементам языка программирования Python (операторы, числовые и строковые переменные, списки, условия и циклы);
– узнать фундаментальные принципы языка Python;
– сформировать навыки самостоятельного решения задач на персональных компьютерах, подбора программных средств, анализа и интерпретация полученных результатов

Обучение по дисциплине «Методы обработки данных с использованием языка программирования Python» направлено на формирование у обучающихся следующих компетенций:

Код и наименование компетенций	**Индикаторы достижения компетенции**
ОПК-3. Способен разрабатывать математические модели и проводить их анализ при решении задач в области профессиональной деятельности	ИОПК-3.1. знает синтаксис и управляющие конструкции языка Python; ИОПК-3.2. знает основные стандартные модули и библиотеки в Python; ИОПК-3.3. знает особенности разработки прикладных программ на языке Python; ИОПК-3.4. знает принципы разработки собственных модулей и библиотек; ИОПК-3.5. знает специализированные библиотеки для анализа данных. ИОПК-3.6. умеет использовать как стандартные, так и дополнительные модули, расширения и пакеты; ИОПК-3.7. умеет создавать собственные модули и импортировать их; ИОПК-3.8. умеет создавать собственные приложения и библиотеки. ИОПК-3.9.владеет навыками разработки приложений с внешними источниками данных (текстовыми файлами, xml-файлами, csv-файлами);

	ИОПК-3.10.владеет практическими навыками использования набора библиотек языка Python для прикладных задач в области анализа данных;

2. Место дисциплины в структуре образовательной программы

Дисциплина относится к обязательной части блока Б1 «Дисциплины (модули)».

Дисциплина «Методы обработки данных с использованием языка программирования Python» взаимосвязана логически и содержательно-методически со следующими дисциплинами:

- Искусственный интеллект в автомобилестроении
- Цифровые технологии в атомобилестроении
- Подключенные автомобили

3. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 8 зачетных единиц (288 часа).

3.1 Виды учебной работы и трудоемкость
(по формам обучения)

3.1.1. Очная форма обучения

№ п/п	Вид учебной работы	Количество часов	Семестры
			1
1	**Аудиторные занятия**	**80**	80
	В том числе:		
1.1	Лекции		
1.2	Семинарские/практические занятия		46
1.3	Лабораторные занятия		32
2	**Самостоятельная работа**	**208**	208
	В том числе:		
2.1	Изучение дополнительной литературы		20
2.2	Подготовка к практическим занятиям		30
2.3	Подготовка к лабораторным работам		30
2.4	Выполнение домашних заданий		20
2.5	Проектная работа		88
3	**Промежуточная аттестация**		
		Экзамен	
	Итого	**288**	

3.2 Тематический план изучения дисциплины
(по формам обучения)

3.2.1. Очная форма обучения

№ п/п	Разделы/темы дисциплины	Трудоемкость, час					
		Всего	Аудиторная работа				Самостоятельная работа
			Лекции	Семинарские/ практические занятия	Лабораторные занятия	Практическая подготовка	
1	**Раздел 1. Введение и базовые понятия**						
1.1	Тема 1. Введение и установка ПО	16		4	2	-	12
1.2	Тема 2. Синтаксис и семантика	16		4	2	-	12
1.3	Тема 3. Типы данных и циклы	16		4	2	-	12
1.4	Тема 4. Функции, классы и модули	18		4	2	-	14
2	**Раздел 2. Модули и пакеты**			4	2		
2.1	Тема 1. Среда разработки (IDE) и пакеты визуализации данных	18		4	2	-	14
2.2	Тема 2. Библиотека NumPy	16		4	2	-	12
2.3	Тема 3. Библиотеки SciPy и SymPy	16		4	2	-	12
2.4	Тема 4. Библиотека Scikit-Learn	16		4	2	-	12
3	**Раздел 3. Разработка ПО**			4			
3.1	Тема 1. Работа с Git и Github	18		4	2	-	14
3.2	Тема 2. Библиотеки PyTest и Sphinx	18		4	2	-	14
3.3	Тема 3. Непрерывная интеграция (CI)	18		4	2	-	14
3.4	Тема 4. Чистый код (Clean code) и Управление разработкой ПО	18		4	2	-	14
4	**Раздел 4. Библиотеки Python для автомобилестроения**						
4.1	Тема 1. Библиотека Python-CAN	24		8	4	-	16
4.2	Тема 2. Библиотека Carla	24		6	4	-	16
Итого		**288**		**46**	**32**	**-**	**208**

3.3 Содержание дисциплины

Раздел 1. Введение и базовые понятия

Тема 1. Введение и установка ПО

Введение, установка Python, Pip и Conda, Работа с файлами требований, Каналы Conda и файл .condarc, IPython, Запуск файлов Python.

Тема 2. Синтаксис и семантика

Ключевые слова, отступ, комментарии, продолжение строки, составные операторы, пробелы, Переменные и указатели, объекты, операторы: Арифметические операторы, Операторы сравнения, Операторы присваивания, Логические операторы, Побитовые операторы, Операторы принадлежности, Операторы тождества.

Тема 3. Типы данных и циклы

Целые числа, вещественные числа, Булевы значения, комплексные числа, строки, списки, кортежи, множества, словари, операции со строками, Условные операторы (if, elif, else), цикл for, цикл for по двум спискам, цикл while.

Тема 4. Функции, классы и модули

Основной синтаксис функций, Аргументы, Аргументы переменной длины, Возврат нескольких значений, Анонимные функции, Аннотации типов, Синтаксис класса, Атрибуты класса, Наследование, Импорт, Импорт с псевдонимом, Импорт конкретного содержимого модуля, Неявный импорт, самостоятельно написанные модули.

Раздел 2. Модули и пакеты

Тема 1. Среда разработки (IDE) и пакеты визуализации данных

Pycharm, Jupyter, установка Jupyter, команды Jupyter, расширения Jupyter, Matplotlib, Основы построения графиков, Построение нескольких графиков на одном, типы графиков, Seaborn, Bokeh, Plotly.

Тема 2. Библиотека NumPy

Импорт Numpy, Создание массивов вручную, Основные атрибуты, конструкторы массивов, Операторы, Универсальные функции, Изменение формы, Индексация и итерация, Нарезка массивов, Складывание массивов, Сравнение, Линейная алгебра, Чтение данных csv.

Тема 3. Библиотеки SciPy и SymPy

Оптимизация, интерполяция, обработка сигналов, линейная алгебра, метод наименьших полных квадратов, интегрирование, решение дифференциальных уравнений, SymPy live, импорт и основы SymPy, встроенное дифференцирование и интегрирование, решение линейной системы уравнений.

Тема 4. Библиотека Scikit-Learn

Общий API scikit-learn, введение в машинное обучение, классификация, регрессия, кластеризация, снижение размерности, выбор модели и предварительная обработка данных, SVM для классификации и регрессии, дерево решений для классификации и регрессии.

Раздел 3. Разработка ПО

Тема 1. Работа с Git и Github

Контроль версий с помощью git, коммит- сообщения, пул-запросы, GitHub, проекты с открытым исходным кодом, структура репозитория, дополнительные файлы для разработчиков, папка с исходным кодом, тесты, файлы конфигурации, проблемы и пул-запросы.

Тема 2. Библиотеки PyTest и Sphinx

Введение Pytest, процессы тестирования программного обеспечения, запуск pytest, тестирование функции, типы утверждений, другие возможности pytest, начало работы с Sphinx, введение в reStructuredText (заголовки, списки, стили текста, изображения, гиперссылки, reStructuredText online), расширение Sphinx autodoc, расширение Sphinx-Gallery.

Тема 3. Непрерывная интеграция (CI)

Требования к CI, CI-решения (Jenkins, Travis CI, GitHub Actions), построение процесса CI в GitHub-Actions, структура.

Тема 4. Чистый код (Clean code) и Управление разработкой ПО

Принципы чистого кода, примеры чистого кода, рефакторинг, форматирование, инструменты, открытый код, внутренний код, изолированное мышление, парадигмы разработки, каскадная модель, Agile.

Раздел 4. Библиотеки Python для автомобилестроения

Тема 1. Библиотека Python-CAN

Установка, конфигурация, API библиотеки, аппаратные интерфейсы, виртуальные интерфейсы, интерфейсы плагинов, скрипты.

Тема 2. Библиотека Carla

Введение, установка, базовые компоненты CARLA, объекты, карты, датчики и данные, трафик, интеграция со сторонними организациями, разработка, пользовательские ресурсы

3.4 Тематика семинарских/практических и лабораторных занятий

3.4.1. Семинарские/практические занятия

Тема 1. Введение и установка ПО
Тема 2. Синтаксис и семантика
Тема 3. Типы данных и циклы
Тема 4. Функции, классы и модули

Тема 5. Среда разработки (IDE) и пакеты визуализации данных
Тема 6. Библиотека NumPy
Тема 7. Библиотеки SciPy и SymPy
Тема 8. Библиотека Scikit-Learn
Тема 9. Работа с Git и Github
Тема 10. Библиотеки PyTest и Sphinx
Тема 11. Непрерывная интеграция (CI)
Тема 12. Чистый код (Clean code) и Управление разработкой ПО
Тема 13. Библиотека Python-CAN
Тема 14. Библиотека Python-CAN
Тема 15. Библиотека Carla
Тема 16. Библиотека Carla

3.4.2. Лабораторные занятия

Тема 1. Визуализация данных в Python
Тема 2. Библиотека NumPy
Тема 3. Библиотеки SciPy и SymPy
Тема 4. Библиотека Scikit-Learn
Тема 5. Библиотеки PyTest и Sphinx
Тема 6. Непрерывная интеграция (CI) и чистый код (Clean code)
Тема 7. Библиотека Python-CAN
Тема 8. Библиотека Carla

3.5 Тематика курсовых проектов (курсовых работ)

Не предусмотрено по учебному плану

4. Учебно-методическое и информационное обеспечение

4.1 Нормативные документы и ГОСТы

http://pm.samgtu.ru/sites/pm.samgtu.ru/files/materials/oop/python3.1.pdf

4.2 Основная литература

1. Программирование компьютерного зрения на языке Python / Ян Эрик Солем; пер. с англ. А.А. Слинкина. – М.: ДМК Пресс, 2016. – 312 с. –URL: https://znanium.com/catalog/product/1027847.

2. Технологии и методы программирования: учебное пособие для прикладного бакалавриата / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — М.:

Юрайт, 2019. – 235 с.: (Серия: Бакалавр. Прикладной курс). – https://biblio-online.ru/bcode/433611.

4.3 Дополнительная литература

1. Жуков, Р. А. Язык программирования Python: практикум : учебное пособие / Р.А. Жуков. — Москва : ИНФРА-М, 2020. — 216 с. + Доп. материалы [Электронный ресурс]. — (Высшее образование: Бакалавриат). — www.dx.doi.org/10.12737/textbook_5cb5ca35aaa7f5.89424805. – URL: https://znanium.com/catalog/product/1045700.

4.4 Электронные образовательные ресурсы

Портал Федеральных государственных образовательных стандартов высшего образования http://fgosvo.ru.
Компьютерная справочная правовая система «КонсультантПлюс» (http://www.consultant.ru/)

4.5 Лицензионное и свободно распространяемое программное обеспечение

- Операционная система Microsoft Windows 10 pro;
- Операционная система macOS Monterey 12.5.1
- Операционная система macOS Catalina 10.15.7
- Операционная система Microsoft Windows Server 2012 R2:
- Программное обеспечение Microsoft Office 365
- Комплексная Система Антивирусной Защиты Kaspersky Total Security для бизнеса Russian Edition
- Файловый архиватор 7z
- Браузеры: Safari, Google Chrome, Mozilla Firefox
- Офисный пакет LibreOffice

4.6 Современные профессиональные базы данных и информационные справочные системы

1. Система федеральных образовательных порталов. Информационнокоммуникационные технологии в образовании. http://www.ict.edu.ru/lib/.
2. Интернет университет информационных технологий. http://www.intuit.ru/.

3. Система федеральных образовательных порталов. Информационнокоммуникационные технологии в образовании. http://www.ict.edu.ru/lib/.

4. Российская национальная библиотека (РНБ). www.hbl-russia.ru.

5. Российская государственная библиотека (РГБ). http://www.rsl.ru.

6. ЭБС «Университетская библиотека онлайн» http:www.biblioclub.ru.

7. ЭБС «Znanium.com» http://znanium.com.

8. ЭБС «Юрайт» https://biblio-online.ru.

5. Материально-техническое обеспечение

Аудитория для проведения учебных занятий (кабинет рекламы и маркетинга). Основное оборудование: доска, учебная мебель, стол, стул преподавателя, флипчарт, расходные материалы (маркеры, бумага для флипчарта) технические средства обучения: персональный компьютер; набор демонстрационного оборудования (проектор, экран, колонки)
Аудитория для проведения учебных занятий (кабинет информатики, математики и 3D графики) оборудование: Основное оборудование: мебель аудиторная, стол преподавателя, стул преподавателя, персональные компьютеры Технические средства обучения: персональный компьютер; набор демонстрационного оборудования (проектор, экран, колонки) Помещение для самостоятельной работы обучающихся Основное оборудование: Оснащены компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа к электронной информационнообразовательной среде

6. Методические рекомендации

6.1 Методические рекомендации для преподавателя по организации обучения

Перед началом изучения дисциплины преподаватель должен ознакомить студентов с видами учебной и самостоятельной работы, перечнем литературы и интернет-ресурсов, формами текущей и промежуточной аттестации, с критериями оценки качества знаний для итоговой оценки по дисциплине.

При проведении лекций преподаватель:

1) формулирует тему и цель занятия;

2) излагает основные теоретические положения;

3) с помощью мультимедийного оборудования и/или под запись дает определения основных понятий, расчетных формул;

4) проводит примеры из отечественного и зарубежного опыта, дает текущие статистические данные для наглядного и образного представления изучаемого материала;

5) в конце занятия дает вопросы для самостоятельного изучения.

При проведении практических занятий преподаватель:

1) формулирует тему и цель занятия;

2) предлагает студентам выполнить контрольные задания, вынесенные на практическое занятие;

3) организует дискуссию по наиболее сложным вопросам;

4) предлагает студентам провести обобщение изученного материала.

В случае проведения аудиторных занятий (как лекций, так и практических занятий) с использованием активных методов обучения (деловых игр, кейсов, мозговых атак, игрового проектирования и др.) преподаватель:

1) предлагает студентам разделиться на группы;

2) предлагает обсудить сформулированные им проблемы согласно теме лекции (практического занятия), раскрывая актуальность проблемы и ее суть, причины, ее вызывающие, последствия и пути решения;

3) организует межгрупповую дискуссию;

4) проводит обобщение с оценкой результатов работы студентов в группах и полученных основных выводов и рекомендаций по решению поставленных проблем. Перед выполнением внеаудиторной самостоятельной работы преподаватель проводит инструктаж (консультацию) с определением цели задания, его содержания, сроков выполнения, основных требований к результатам работы, критериев оценки, форм контроля и перечня источников и литературы.

Для оценки полученных знаний и освоения учебного материала по каждому разделу и в целом по дисциплине преподаватель использует формы текущего, промежуточного и итогового контроля знаний обучающихся.

6.2 Методические указания для обучающихся по освоению дисциплины

ОБЩИЕ РЕКОМЕНДАЦИИ

Успешное освоение курса предполагает активное, творческое участие студента на всех этапах ее освоения путем планомерной, повседневной работы.

При подготовке к аудиторным занятиям, непосредственно в ходе проведения лекций и практических занятий, а также в ходе самостоятельной работы студенты должны пользоваться учебной литературой (согласно утвержденному перечню основной и дополнительной литературы по данному курсу), учебно-методическими материалами (включая данную рабочую

программу), которые размещены в электронной информационно-образовательной среде.

ИЗУЧЕНИЕ ТЕОРЕТИЧЕСКОГО КУРСА

Правила конспектирования

Конспект является письменным текстом, в котором кратко и последовательно изложено содержание основного источника информации. Конспектировать – значит приводить к некоему порядку сведения, почерпнутые из оригинала. В основе процесса лежит систематизация прочитанного или услышанного. Записи могут делаться как в виде точных выдержек, цитат, так и в форме свободной подачи смысла.

Манера написания конспекта, как правило, близка к стилю первоисточника. Если конспект составлен правильно, он должен отражать логику и смысловую связь записываемой информации.

В хорошо сделанных записях можно с легкостью обнаружить специализированную терминологию, понятно растолкованную и четко выделенную для запоминания значений различных слов. Используя законспектированные сведения, легче создавать значимые творческие или научные работы, различные рефераты и статьи.

САМОПОДГОТОВКА К ТЕКУЩЕМУ КОНТРОЛЮ

Просмотрите конспект сразу после занятий. Пометьте материал конспекта лекций, который вызывает затруднения для понимания. Попытайтесь найти ответы на затруднительные вопросы, используя предлагаемую литературу. Постарайтесь разобраться с непонятным материалом, в частности новыми терминами. Часто незнание терминологии мешает воспринимать материал на теоретических и лабораторно-практических занятиях на должном уровне. Если самостоятельно не удалось разобраться в материале, сформулируйте вопросы и обратитесь на текущей консультации или на ближайшей лекции за помощью к преподавателю. Каждую неделю рекомендуется отводить время для повторения пройденного материала, проверяя свои знания, умения и навыки по контрольным вопросам. Ответьте на контрольные вопросы для самопроверки, имеющиеся в учебнике или предложенные в данных методических рекомендациях. Кратко перескажите содержание изученного материала «своими словами». Заучите «рабочие определения» основных понятий, законов. Освоив теоретический материал, приступайте к выполнению заданий, упражнений; решению задач, расчетов самостоятельной работы, составлению графиков, таблиц и т.д.

7. Фонд оценочных средств

7.1 Методы контроля и оценивания результатов обучения

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

- подготовка и выполнение лабораторных работ;
- выполнение контрольный заданий.

7.2 Шкала и критерии оценивания результатов обучения

Шкала оценивания	Описание
Отлично	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Хорошо	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует неполное, правильное соответствие знаний, умений, навыков приведенным в таблицах показателей, либо если при этом были допущены 2-3 несущественные ошибки.
Удовлетворительно	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность.
Неудовлетворительно	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует отсутствие знаний, умений, навыков по ряду показателей.

7.3 Оценочные средства

7.3.1. Текущий контроль

Порядок выполнения контрольной работы

Приступать к написанию контрольной работы следует лишь после изучения основных тем дисциплины, основываясь на учебнике (учебном пособии) из списка основной литературы, рекомендованной по данному курсу, дополнительной учебной литературы, и нормативного материала по избранной теме (при наличии такой необходимости). Список рекомендуемой преподавателем научной литературы необходимо рассматривать как основу для самостоятельного поиска и анализа.

Подбор материала и план контрольной работы разрабатывается студентом самостоятельно, что дает преподавателю основание оценить степень усвоения изученного материала. При написании контрольной работы студенту следует

проявить самостоятельность и не прибегать к простому переписыванию литературы. Преподаватель вправе учитывать качество проделанной работы при сдаче студентом зачета или экзамена по соответствующей дисциплине.

Введение контрольной работы должно содержать формулировку контрольного задания, краткое изложение цели контрольной работы.

Основная часть контрольной работы должна содержать базовые определения, доказательства, описание методики расчётов. В ходе написания основной части следует давать ссылки на используемые источники информации. В этой части следует также изложить ход собственных рассуждений, описать последовательность расчётов, привести промежуточные доказательства и результаты решения поставленной задачи.

В заключении следует сформулировать краткие выводы по проделанной работе и привести список использованных источников информации.

Типичными ошибками при выполнении контрольной работы являются:

● несоответствие содержания контрольной работы цели и поставленным задачам;

● неверное решение предложенных задач;

● нарушение установленных требований к оформлению работы;

● использование информации без ссылок на источник информации.

ПОДГОТОВКА К ПРАКТИЧЕСКИМ ЗАНЯТИЯМ

Как готовиться к практическому занятию.

1. Внимательно прочтите вопросы к заданию.

2. Подберите литературу, не откладывайте ее поиски на последний день.

3. Прочтите указанную литературу, определите основной источник по каждому вопросу, делая выписки на листах или карточках, нумеруйте их пунктами плана, к которому они относятся.

4. Оформляя выписки, не забудьте записать автора, название, год и место издания, том, страницу.

5. При чтении найдите в словарях значение новых слов или слов, недостаточно вам известных.

6. Просматривая периодическую печать, делайте вырезки по теме.

7. Проверьте, на все ли вопросы плана у вас есть ответы.

8. На полях конспекта, выписок запиши вопросы, подчеркните спорные положения в тексте.

Требования к выступлению

Перечень требований к любому выступлению студента примерно таков: - связь выступления с предшествующей темой или вопросом; - раскрытие сущности проблемы; - методологическое значение для научной, профессиональной и практической деятельности. Важнейшие требования к выступлениям студентов – самостоятельность в подборе фактического материала и аналитическом отношении к нему, умение рассматривать примеры и факты во взаимосвязи и взаимообусловленности, отбирать наиболее существенные из них. Приводимые участником практического занятия примеры и факты должны быть существенными, по возможности перекликаться с

профилем обучения. Примеры из области наук, близких к будущей специальности студента, из сферы познания, обучения поощряются руководителем семинара. Выступление студента должно соответствовать требованиям логики. Четкое вычленение излагаемой проблемы, ее точная формулировка, неукоснительная последовательность аргументации именно данной проблемы, без неоправданных отступлений от нее в процессе обоснования, безусловная доказательность, непротиворечивость и полнота аргументации, правильное и содержательное использование понятий и терминов.

7.3.2. Промежуточная аттестация

Примерные вопросы на экзамен:

1. Язык Python и особенности его стиля программирования. Интерактивный режим Python.
2. Синтаксис и управляющие конструкции языка Python. Переменные, значения и их типы. Типы данных в Python.
3. Встроенные операции и функции. Основные алгоритмические конструкции.
4. Условный оператор. Множественное ветвление.
5. Циклы и счетчики.
6. Определение функций. Параметры и аргументы. Вызовы функций. Оператор возврата. Конструкции *args, **kwargs.
7. Списки, кортежи и словари.
8. Операторы общие для всех типов последовательностей.
9. Специальные операторы и функции для работы со списками. Срезы.
10. Работа со словарями. Методы словарей.
11. Случайные числа. random, randrange, choice.
12. Функции обработки строк. join, replace, split.
13. Стандартная библиотека и pip. Модули и пакеты в Python. Основные стандартные модули.
14. Импортирование модулей. Создание собственных модулей и их импортирование. Специализированные модули и приложения.
15. Файлы и исключения. Работа с внешними источниками данных.
16. Исключения, обработка исключений, вызов исключений (try-except-finally).
17. Утверждения (assert). Открытие, чтение, запись. (open, инструкция with).
18. Работа с текстовыми файлами, xml и csv - файлами.

19. Функциональное программирование. Лямбда-функции.
20. Использование функций map, filter, reduce, zip.
21. Генераторы, декораторы, рекурсия.
22. Модификация функций с помощью декораторов.
23. Итерируемые объекты. Использование генераторов (yield).
24. ООП в Python. Классы, объекты и экземпляры классов. Наследование.
25. Магические методы. Переопределение операторов. Методы классов.
26. Инкапсуляция. Условно частные и строго частные методы.
27. Регулярные выражения. Использование регулярных выражений. Пакет re.
28. Наука о данных и Python. Библиотеки: NumPy, pandas, matplotlib, SciPy.
29. Основы NumPy: массивы и векторные вычисления.
30. Инструменты визуализации данных для Python.
31. Введение в API библиотеки matplotlib.
32. Библиотека pandas. Введение в структуры данных pandas.
33. Объекты Dataframe и Series.
34. Визуализация данных в pandas. Seaborn.
35. Агрегирование данных и групповые операции.
36. Научные и инженерные расчеты с использованием библиотеки scipy.