

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

УТВЕРЖДАЮ

Проректор по учебной работе и
образовательным инновациям

_____ О.Г. Прохоренко

«05» июля 2023 г.

Регистрационный № УД – 612/б.



КОМПЬЮТЕРНАЯ МАТЕМАТИКА

**Учебная программа учреждения высшего образования
по учебной дисциплине для специальности:**

6-05-0533-08 Компьютерная математика и системный анализ

2023 г.

Учебная программа составлена на основе образовательного стандарта специальности 6-05-0533-08 «Компьютерная математика и системный анализ» ОСВО 6-05-0533-08-2023, учебных планов БГУ № 6-5.4-56/01 от 15.05 2023, № 6-5.4-56/11 ин. от 31.05.2023.

СОСТАВИТЕЛИ:

Л.Л. Голубева, заведующая кафедрой дифференциальных уравнений и системного анализа механико-математического факультета Белорусского государственного университета, кандидат физико-математических наук, доцент;

О.А. Лаврова, доцент кафедры дифференциальных уравнений и системного анализа механико-математического факультета Белорусского государственного университета, кандидат физико-математических наук;

А.Э. Малевич, доцент кафедры дифференциальных уравнений и системного анализа механико-математического факультета Белорусского государственного университета, кандидат физико-математических наук, доцент;

Н.Л. Щеглова, доцент кафедры дифференциальных уравнений и системного анализа механико-математического факультета Белорусского государственного университета, кандидат физико-математических наук, доцент.

РЕЦЕНЗЕНТЫ:

Т.Н. Жоровина, доцент кафедры теории функций механико-математического факультета Белорусского государственного университета, кандидат физико-математических наук, доцент;

Е.А. Крушевский, доцент кафедры математические методы в строительстве Белорусского национального технического университета, кандидат физико-математических наук, доцент.

РЕКОМЕНДОВАНА К УТВЕРЖДЕНИЮ:

Кафедрой дифференциальных уравнений и системного анализа БГУ (протокол № 13 от 14.06.2023)

Научно-методическим советом БГУ
(протокол № 9 от 29.06.2023)

Зав. кафедрой дифференциальных уравнений
и системного анализа, доцент



Л. Л. Голубева

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

В настоящее время бурное развитие получило новое, актуальное научное направление – компьютерная математика. Дисциплина «Компьютерная математика» может быть определена как обучающий курс, представляющий совокупность фундаментальных знаний, методов, практик, алгоритмических и компьютерных средств, предназначенных для эффективного решения всех видов задач математического содержания, с высокой степенью визуализации результатов на всех этапах исследований.

Подход к решению задач компьютерными методами заключается в использовании готовых математических сред для исследований – систем компьютерной математики, таких как: MathCAD, MATLAB, *Mathematica*, Maple, Python и других, с использованием возможности их интеграции.

Системы компьютерной математики позволяют исследовать проблему, провести анализ и визуализацию данных, построить модели, их тестировать, проверить существование решения, документировать и оформить результаты. При их использовании основное внимание сосредотачивается на существовании проблемы и компьютерных моделях для ее разрешения, оставляя в стороне технические вопросы классической математики, детали вычислительных методов и алгоритмических процедур, нюансы языков программирования и команд операционной системы.

Цели и задачи учебной дисциплины

Цель учебной дисциплины – повышение уровня профессиональной компетентности студентов путем обучения навыкам и умениям компьютерного моделирования и проведения исследований для решения проблем классического и современного естествознания.

Образовательная цель: обучение студентов методам и приемам компьютерного моделирования в современных компьютерных математических средах, эффективному исследованию посредством компьютера широкого круга проблем математического содержания.

Развивающая цель: формирование у студентов умений использования существующих и самостоятельной разработки новых технологий компьютерного моделирования.

Задачи учебной дисциплины:

- исследование и разработка моделей, алгоритмов, методов, программных решений, инструментальных средств по тематике выполняемых научно-исследовательских проектов;
- разработка и исследование математических, информационных и имитационных моделей в определенной предметной области;
- составление и подготовка к публикации научных обзоров, рефератов, библиографий, научно-технических отчетов по тематике проводимых исследований.

Место учебной дисциплины в системе подготовки специалиста с высшим образованием.

Учебная дисциплина относится к модулю «Компьютерная математика» государственного компонента.

При изучении дисциплины «Компьютерная математика» используются знания, умения и навыки, полученные при изучении дисциплин «Математический анализ» модуля «Основы анализа», «Дифференциальные уравнения» модуля «Дифференциальные уравнения» государственного компонента, дисциплин модуля «Алгебра и геометрия», дисциплины «Методы программирования», «Технологии программирования» модуля «Программирование» компонента учреждения высшего образования.

Требования к компетенциям

Освоение учебной дисциплины «Компьютерная математика» должно обеспечить формирование у студентов следующих универсальных и базовых профессиональных компетенций:

универсальные компетенции:

УК-2. Решать стандартные задачи профессиональной деятельности на основе применения информационно-коммуникационных технологий;

базовые профессиональные компетенции:

БПК-3. Применять математический аппарат в интеграции с компьютерными средами для создания и исследования моделей различных уровней абстракции.

В результате изучения дисциплины «Компьютерная математика» студент должен для каждой из современных компьютерных математических систем *Mathematica*, MATLAB, а также для среды разработки JupyterLab

знать:

- идеологию системы и принципы работы в ней; инструментальные средства, элементы управления, интерфейс; структуры данных; особенности построения функций пользователя; возможности визуализации исследований и оформления результатов исследований в виде публикаций;
- особенности языков программирования в среде каждого пакета;

уметь:

- применять современный математический аппарат в эффективной интеграции с инструментальными компьютерными математическими средствами;
- создавать и исследовать математические, компьютерные, имитационные модели различных уровней абстракции;
- разрабатывать и анализировать алгоритмы, методы и программные решения по тематике выполняемых исследований;
- квалифицированно применять языки программирования современных систем компьютерной математики;
- проводить анализ результатов исследований, строить информационные модели в средах современных математических пакетов;

- готовить материалы к публикации, в том числе в электронных изданиях, по тематике и результатам проводимых исследований;
- самостоятельно расширять компьютерные математические знания с дальнейшим их использованием при построении и анализе математических и компьютерных моделей широкого круга теоретических и прикладных задач.

владеть:

- методами и приемами построения моделей объектов, данных, процессов, систем;
- методами исследований и решения проблем математического содержания с использованием математических компьютерных приложений.

Структура учебной дисциплины

Дисциплина «Компьютерная математика» изучается в 1-м, 2-м и 3-м семестрах. Форма получения высшего образования очная (дневная).

Всего на изучение учебной дисциплины «Компьютерная математика» отводится 310 часов, в том числе 176 аудиторных часов, из них: лекции – 88 часов, лабораторные занятия – 72 часа, управляемая самостоятельная работа – 16 часов. Трудоемкость учебной дисциплины составляет 9 зачетных единиц.

В 1 семестре: всего 110 часов, в том числе 72 аудиторных часа, из них: лекции – 36 часов (в том числе – 4 ч/ДОТ), лабораторные занятия – 30 часов (в том числе 4 ч/ДОТ), управляемая самостоятельная работа – 6 часов.

Трудоемкость учебной дисциплины составляет 3 зачетные единицы.

Форма промежуточной аттестации – зачет и экзамен.

Во 2 семестре: всего 110 часов, в том числе 68 аудиторных часов, из них: лекции – 34 часа, лабораторные занятия – 28 часов, управляемая самостоятельная работа – 6 часов.

Трудоемкость учебной дисциплины составляет 3 зачетные единицы.

Форма промежуточной аттестации – зачет и экзамен.

В 3 семестре: всего 90 часов, в том числе 36 аудиторных часов, из них: лекции – 18 часов (в том числе – 4 ч/ДОТ), лабораторные занятия – 14 часов (в том числе – 4 ч/ДОТ), управляемая самостоятельная работа – 4 часа.

Трудоемкость учебной дисциплины составляет 3 зачетные единицы.

Форма промежуточной аттестации – экзамен.

СОДЕРЖАНИЕ УЧЕБНОГО МАТЕРИАЛА

Раздел 1. Символьный пакет *Mathematica*

Тема 1.1. Сценарий работы в среде символьного математического пакета

Позиционирование дисциплины Компьютерная Математика в системе математических дисциплин. Моделирование как метод научного познания. Особенности компьютерного моделирования. Компьютерные математические среды для моделирования объектов, явлений, процессов.

Числовые и символьные пакеты. Назначение и структура символьных математических пакетов. Интерфейс. Сценарий работы. Сессия (*Session*) как объект работы Ядра (*Kernel*). Накопление знаний во время Сессии и хранение знаний между сессиями и сеансами работы с пакетом. Метафора записной книжки (*Notebook*). Ячейки (*Cells*) как основные обобщенные объекты Блокнотов. Основной цикл ($In[xx] \rightarrow Out[xx]$), протокол *MathLink* работы пакета.

Оформление электронного документа. Выявление свойств Гамма-функции.

Тема 1.2. Выражение как основная структура данных

Понятие рекурсии. Рекурсивное определение выражения. Анализ структуры выражения. Атомарные объекты. Функции-конверторы типов атомарных объектов. Оператор *Apply*. Список как наиболее используемое выражение. Встроенные функции для работы со списками.

Исследование структуры выражения.

Тема 1.3. Образцы

Образец (*Patterns*) как мета выражение. Основные правила записи. Описание различных множеств посредством использования образцов. Образцы, соответствующие условию. Образцы, содержащие альтернативу. Установка значения выражения по умолчанию. Образцы, допускающие повторение выражений. Встроенное значение части выражения. Построение функции пользователя: образцы при описании ОДЗ функции.

Построение функций пользователя, исследующих структуру выражения.

Тема 1.4. Функциональный стиль программирования

Чистые, или анонимные функции. Семейство операторов *Map*. Возможности последовательного применения функций, операторы повторного действия (*Nest*, *Fold*, *FixedPoint*). Рекурсивное задание функций.

Алгебраические преобразования. Метод неопределенных коэффициентов в задаче нахождения частного и остатка от деления многочленов одной переменной, его реализация с использованием анонимных функций. Построение суперпозиции чистых функций.

Тема 1.5. Основы графики

Графические примитивы. Режимы выполнения графических функций. Базовые функции двумерной и трехмерной графики. Специализированные графические пакеты.

Проектирование функций, осуществляющих преобразование множества точек комплексной плоскости. Координатные сетки на плоскости: прямоугольная декартова, полярная. Роль и задание цвета при построении сетки. Отображение графического объекта на плоскости, состоящего из точек и линий, при действии на него заданной функцией.

Тема 1.6. Правила преобразований. Глобальные определения

Функции семейства `Set`. Механизм верхних и нижних значений символа. Программирование, основанное на глобальных правилах преобразований. Способы задания, представления и отображения функций. Восстановление функции по заданным свойствам.

Восстановление Гамма-функции по ее свойствам и значениям в узлах на заданном отрезке.

Тема 1.7 Локальные правила преобразований, подстановки

Определение локальных правил преобразований. Использование образцов. Функции семейства `Replace`. Подстановки в тождественных преобразованиях алгебраических выражений. Программирование, основанное на локальных правилах преобразований.

Подстановки. Построение псевдо- n -мерного гиперкуба и геометрических фракталов.

Тема 1.8 Построение системы «Аналитическая геометрия на плоскости»

Проектирование базовой системы «Аналитическая геометрия на плоскости»: принципы работы и общие соглашения. Представление объектов «точка», «вектор», «прямая» в компьютерной среде *Mathematica*. Проектирование и реализация способов задания геометрических объектов. Построение операторов, извлекающих свойства геометрических объектов. Создание функций, строящих графические образы геометрических объектов.

Анализ и модификация базовых объектов системы «Аналитическая геометрия на плоскости». Построение печатных образов объектов. Присвоение имен.

Реализация системы «Аналитическая геометрия на плоскости».

Тема 1.9. Визуализация исследований. Динамическая интерактивность

Вычисление выражений внутри функций `Dynamic` и `Manipulate`. Интерактивная графика. Построение демонстрационных моделей.

Решение задач геометрии на плоскости с использованием возможностей системы «Аналитическая геометрия на плоскости» и динамической интерактивности.

Тема 1.10. Проектирование системы «Аналитическая геометрия» в n -мерном пространстве

Внутреннее представление базовых объектов «точка в пространстве», «вектор в пространстве», «прямая в пространстве», «плоскость». Проектирование и реализация способов задания базовых объектов. Построение операторов, извлекающих свойства базовых объектов. Создание функций, строящих образы базовых объектов. Анализ построенной системы и обобщение основных положений для n -мерного пространства.

Частичная реализация системы «Аналитическая геометрия в пространстве».

Тема 1.11. Организация вычислительного процесса в *Mathematica*

Конструкции, управляющие ходом вычисления: операторы выбора и операторы повторения.

Роль символа в процессе вычислений. Предопределенные константы. Атрибуты символа. Опции, или режимы выполнения функций. Глобальные определения правил преобразований.

Порядок вычисления выражения. Стандартный процесс вычисления выражения. Выражения, вычисляемые нестандартно. Вычисление правил преобразований, глобальных определений. Изменение стандартного порядка вычисления выражения в структурах выбора и повторения.

Тема 1.12. Принципы локализации переменных. Аппроксимация Паде

Механизмы локализации символов в определенных границах. Итераторы. Функции Block, Module, With. Механизм контекстов.

Понятие аппроксимации Паде для функции одной переменной. Построение многочленов Тейлора функции одной переменной, при различных способах задания функции: явное задание функции, функция задана посредством задачи Коши, функция задана неявно. Проектирование функций для построения аппроксимации Паде, использование метода неопределенных коэффициентов. Особенности реализации аппроксимации Паде при различных способах задания функции. Визуализация заданной функции, ее многочленов Тейлора и ее аппроксимации Паде.

Раздел 2. Компьютерное моделирование на Python

Тема 2.1. Язык Python. Синтаксические особенности языка. Среда разработки JupyterLab

Основные характеристики языка программирования Python. Применение языка программирования Python. Структура программы на Python. Байт-код. Виртуальная машина Python. Отличие синтаксиса Python от синтаксиса других современных языков программирования. Особенности синтаксиса. Способы выполнения программ, написанных на Python.

Принципы работы с интерактивными документами Jupyter Notebook (ipynb файлы) в среде разработки JupyterLab. Типы ячеек, режимы работы с ячейками. Ядро Python.

Тема 2.2. Объект как фундаментальное понятие в Python. Встроенные типы объектов и операции над ними

Понятие объекта. Атрибуты и методы объекта. Встроенные типы числовых объектов (int, float, complex), встроенный булевый тип (bool), встроенные типы для объектов последовательностей (string, list, tuple), встроенный тип отображение (dict), встроенный тип множество (set). Изменяемые и неизменяемые типы объектов.

Переменная, значение переменной, ссылка на объект. Динамическая типизация переменных. Копирование объектов. Операции и методы для

встроенных типов объектов. Выражение спискового включения. Создание строкового объекта на основе подстановок (форматирование строк). Файловый объект.

Тема 2.3. Операторы

Синтаксические правила для составного и составного операторов. Операторы присваивания. Типы присваиваний: базовое, присваивание последовательностей, групповое, дополненное, расширенное. Неявные присваивания. Пространство имен. Оператор выбора. Оператор сопоставления. Операторы цикла. Область видимости переменной. Функция вывода.

Тема 2.4. Функция. Область видимости переменной

Определение функции: операторы `def` и `return`, значения по умолчанию для аргументов, произвольное количество аргументов, порядок следования аргументов. Вызов функции: позиционные и ключевые аргументы. Передача значений аргументов через неявное присваивание. Режимы сопоставления объектов-значений и аргументов-переменных при вызове функции. Объект функции. Определение и применение `lambda`-функций.

Типы областей видимости переменной: локальная область видимости, область видимости объемлющих функций, глобальная область видимости, встроенная область видимости. Правило LEGB для распознавания области видимости имени переменной. Доступ к пространству имен. Операторы объявления `global`, `nonlocal`.

Правила оформления строк документации для функций. Аннотация функции.

Анимация движения секущей прямой к заданной линии на плоскости. Пользовательская функция для моделирования границы правильного многоугольника Рело. Анимация качения треугольника Рело.

Тема 2.5. Механизм итераций

Итерируемые объекты. Проход по элементам итерируемого объекта. Объект итератора. Итерационные инструменты. Протокол итераций. Генераторные объекты. Генераторные выражения. Генераторные функции.

Вложенные итерируемые объекты. Задача о выравнивании списка, состоящего из итерируемых объектов.

Тема 2.6. Функциональное программирование

Инструменты функционального программирования. Функции `map`, `filter`, `enumerate`, `zip`. Функция `reduce` из модуля `functools`. Фабричная функция, функция-замыкание, декоратор функции.

Создание текстового файла большого объема с помощью пользовательской генераторной функции.

Тема 2.7. Модуль как важнейшая программная структура

Понятие модуля, пакета, расширения. Модульная структура программ. Импорт модуля: загрузка файла, создание объекта модуля, выполнение байт-кода интерпретатором. Операторы `import` и `from`. Объект модуля. Доступ к переменным импортированного модуля. Перегрузка модуля. Стандартная библиотека модулей Python.

Тема 2.8. Основы объектно-ориентированного программирования

Понятия класса и экземпляра класса. Оператор создания класса. Методы класса. Метод инициализации экземпляра класса. Дерево наследования. Настройка кода. Создание классов на основе встроенных типов. Перегрузка операций и встроенных функций.

Бинарное дерево поиска. Представление. Проектирование классов. Построение бинарного дерева поиска. Перегрузка операции принадлежности `in`. Перегрузка встроенной функции `len`. Обходы бинарного дерева поиска. Центрированный обход. Поиск минимального и максимального значения в дереве.

Проектирование классов. Композиция. Делегирование. Абстрактный класс. Синглтон. Классовые методы. Статические методы. Атрибут-свойство.

Тема 2.9. Пакет `matplotlib` для графического представления данных

Структура пакета `matplotlib`. Создание графических окон и графических областей. Процедурный подход для построения двумерных графиков: оформление графика, текстовые объекты в графической области. Объектно-ориентированный подход для построения изображений. Графические объекты. Методы графических объектов.

Тема 2.10. Расширение `numpy` для численных вычислений

Тип данных массив (`ndarray`) как базовая структура данных расширения `numpy`. Создание массивов. Доступ к элементам и группе элементов массивов. Изменение структуры массивов. Векторизация вычислений с массивами. Операции линейной алгебры. Модуль `linalg`.

Тема 2.11. Программирование с использованием исключений

Изменение потока управления с помощью исключений. Стандартный обработчик исключений. Встроенные исключения. Перехват исключений с помощью оператора `try`. Генерация исключений интерпретатором Python или с помощью оператора `raise`. Оператор `assert` для контроля за ограничениями на переменные, определяемые пользователем. Диспетчер контекстов. Примеры использования исключений.

Раздел 3. Числовой пакет MATLAB

Тема 3.1. Назначение, характеристика и структура пакета MATLAB. Интерфейс пользователя

Назначение и области применения системы MATLAB. Операционная среда и язык программирования системы MATLAB. Концепция Пакетов Прикладных Программ. Сравнительная характеристика пакета MATLAB и других числовых и символьных математических пакетов. Структура пакета MATLAB. Начало работы с MATLAB. Сессия работы с MATLAB. Интерфейс пользователя. Работа в режиме командной строки. Сообщения об ошибках. Интерактивная справка из командной строки. Справочная система и демонстрационные примеры.

Тема 3.2. Встроенные типы данных. Векторы, матрицы, многомерные массивы

Основные объекты в MATLAB. Математические выражения, константы, действительные и комплексные числа, системные переменные. Форматы чисел. Переменные. Иерархия типов данных в MATLAB. Вектор, матрица, тензор. Операции с векторами и матрицами. Операции конкатенации, индексации и формирования диапазона значений. Работа со строками и столбцами. Многомерные массивы. Доступ к элементам и группе элементов массивов. Операторы и встроенные функции. Операции отношения и логические операции. Символьные массивы. Структуры и массивы структур. Массивы ячеек. Таблицы.

Тема 3.3. Элементы программирования

Управляющие структуры. Операторы управления. Операторы ветвления. Операторы цикла. Обработка исключительных ситуаций.

Тема 3.4. Сценарии, функции, переменные

Функции пользователя: inline-функции, анонимные функции. Понятие М-сценария и М-функции. Структура, свойства и выполнение М-сценария. Разбиение М-сценария на секции. Выполнение отдельной команды, группы команд, секции М-сценария. Структура М-функции. Справка `help` по объекту. Справка `lookfor` по ключевому слову. Синтаксис определения и вызова М-функций. Проверка входных параметров и выходных значений М-функций. Видимость имен переменных и имен функций. Подфункции. Каталоги `private`. Функции с переменным числом аргументов. Локальные и глобальные переменные. Разработка и отладка М-функций. Комментарии.

Работа с программными файлами в режиме EDITOR. Работа с программными файлами в режиме LIVE EDITOR. Инструмент LIVE EDITOR TASK.

Тема 3.5. Основы объектно-ориентированного программирования, классы и объекты

Принципы объектно-ориентированного программирования (ООП), реализуемые в системе MATLAB: инкапсуляция, одиночное и множественное наследование, агрегирование. Понятие класса и объекта. Основные правила определения класса и создания объекта. Функция-конструктор для создания объекта класса. Проверка принадлежности объекта заданному классу. Свойства и методы класса. Переопределение встроенных функций и операторов для классов пользователя.

Однофайловый и мультифайловый способы создания классов пользователя.

Тема 3.6. Импорт и экспорт данных

Данные `mat`-формата. Сохранение и восстановление данных рабочего пространства. Поддерживаемые в MATLAB форматы файлов. Файловый ввод-вывод. Файловые операции с бинарными данными. Файловые операции с форматированными данными. Позиционирование файла. Работа с текстовыми файлами. Работа с графическими файлами. Работа с электронными таблицами. Процесс импорта данных через инструмент Import Tool.

Тема 3.7. Высокоуровневая и дескрипторная графика

Графическое окно `Figure` и координатные оси `Axes`. Интерфейс графических окон. *Двумерная графика*. Изменение свойств линий. Управление свойствами осей графиков. Графики в декартовой и полярной системах координат. *Анимационная графика*. Движение точки на плоскости и в пространстве.

Трёхмерная графика. Линии и поверхности. Положение камеры и графического объекта. Цветовая палитра. Прозрачность графических объектов. Контурные графики.

Текстовое оформление графиков. Титульная и осевые надписи, легенда. Вывод текста в заданное место графика. Форматирование графиков со встроенным графическим редактором MATLAB. Сохранение графических изображений.

Дескрипторная графика Handle Graphics. Иерархия графических объектов. 2D- и 3D-графические объекты. Дескрипторы графических объектов. Функции создания и управления свойствами графических объектов. Управление свойствами по умолчанию графических объектов.

Тема 3.8. Средства создания приложений с графическим интерфейсом пользователя

Понятие графического интерфейса пользователя GUI. Графические элементы управления системы MATLAB. Способы создания графического интерфейса пользователя при разработке приложений.

Программирование GUI с использованием функций MATLAB и (или) графических компонентов UI figures пользовательского интерфейса. Специальные функции MATLAB для работы с графическими объектами. Callback-функции для обработки событий.

Визуальное проектирование графического интерфейса пользователя с использованием приложения App Designer. Компоненты для создания GUI.

Визуальное проектирование графического интерфейса пользователя с использованием приложения GUIDE. Браузер объектов управления Object Browser. Инспектор свойств объекта Property Inspector.

Тема 3.9. Вычисления в MATLAB. Обработка данных

Исследование функций. Нахождение нулей функции $y=f(x)$. Нахождение нулей функции, заданной полиномом. Поиск локальных минимумов функций одной и двух переменных.

Задачи линейной алгебры. Представление систем линейных уравнений в векторно-матричном виде. Решение СЛУ. Метод Крамера решения СЛУ.

Интерполяция и аппроксимация данных. Интерполяция одномерных данных полиномиальной функцией по методу наименьших квадратов. Сплайновая интерполяция функциями одной и двух переменных.

УЧЕБНО-МЕТОДИЧЕСКАЯ КАРТА УЧЕБНОЙ ДИСЦИПЛИНЫ

Очная форма получения высшего образования с применением дистанционных образовательных технологий (ДОТ)

Номер раздела, темы	Название раздела, темы	Количество аудиторных часов					Количество часов УСР	Форма контроля знаний
		Лекции	Практические занятия	Семинарские занятия	Лабораторные занятия	Иное		
1	2	3	4	5	6	7	8	9
1.	Символьный пакет Mathematica	36			30		6	
1.1	Сценарий работы в среде символьного математического пакета	2			2			Устный опрос
1.2	Выражение как основная структура данных	2			2			Отчет по лабораторной работе с устной защитой. Контрольная работа
1.3	Образцы	2 (ДОТ)			2 (ДОТ)			Отчет по лабораторной работе с устной защитой, устный опрос.

1.4	Функциональный стиль программирования	4			4		2	Отчет по лабораторной работе с устной защитой. Отчет по заданиям с устной защитой. Контрольная работа.
1.5	Основы графики	2 (ДОТ)			2 (ДОТ)			Отчет по лабораторной работе с устной защитой, устный опрос.
1.6	Правила преобразований. Глобальные определения	2			2			Отчет по лабораторной работе с устной защитой
1.7	Локальные правила преобразований, подстановки	2			2			Отчет по лабораторной работе с устной защитой
1.8	Построение системы «Аналитическая геометрия на плоскости»	8			6		2	Отчет по лабораторной работе с устной защитой. Отчет по заданиям с устной защитой. Контрольная работа.
1.9	Визуализация исследований. Динамическая интерактивность	3			2			Отчет по лабораторной работе с устной защитой, устный опрос.
1.10	Проектирование системы «Аналитическая геометрия» в n -мерном пространстве, $n > 2$	3			2			Отчет по лабораторной работе с устной защитой.

1.11	Организация вычислительного процесса в <i>Mathematica</i>	2					Устный опрос
1.12	Принципы локализации переменных. Аппроксимация Паде	4			4	2	Отчет по лабораторной работе с устной защитой. Отчет по заданиям с устной защитой. Контрольная работа
2.	Компьютерное моделирование на Python	34			28	6	
2.1	Язык Python. Синтаксические особенности языка. Среда разработки JupyterLab	4			4		Отчет по лабораторной работе с устной защитой
2.2	Объект как фундаментальное понятие в Python. Встроенные типы объектов и операции над ними	4			2	2	Отчет по лабораторной работе с устной защитой. Контрольная работа
2.3	Операторы	2			2		Отчет по лабораторной работе с устной защитой
2.4	Функция. Область видимости переменной	4			4		Отчет по лабораторной работе с устной защитой
2.5	Механизм итераций	2			4		Отчет по лабораторной работе с устной защитой. Контрольная работа
2.6	Функциональное программирование	2			4	2	Отчет по лабораторной работе с устной защитой. Контрольная работа

2.7	Модуль как важнейшая программная структура	2					Устный опрос
2.8	Основы объектно-ориентированного программирования	8		6		2	Отчет по лабораторной работе с устной защитой. Контрольная работа
2.9	Пакет matplotlib для графического представления данных	2		2			Отчет по лабораторной работе с устной защитой.
2.10	Расширение numpy для численных вычислений	2					Устный опрос
2.11	Программирование с использованием исключений	2					Контрольный опрос
3.	Числовой пакет MATLAB	18		14		4	
3.1	Назначение, характеристика, структура пакета MATLAB. Интерфейс пользователя	2		2			Отчет по лабораторной работе с устной защитой, устный опрос.
3.2	Встроенные типы данных. Векторы, матрицы, многомерные массивы	3		3			Отчет по лабораторной работе с устной защитой. Контрольная работа.
3.3	Элементы программирования	1		1			Контрольный опрос
3.4	Сценарии, функции, переменные	2 (ДОТ)		2 (ДОТ)			Отчет по лабораторной работе с устной защитой. Контрольная работа.
3.5	Основы объектно-ориентированного программирования, классы и объекты	2				2	Отчет по заданиям с устной защитой.

3.6	Импорт и экспорт данных	2 (ДОТ)			2 (ДОТ)			Отчет по лабораторной работе с устной защитой, устный опрос
3.7	Высокоуровневая и дескрипторная графика	2			2			Отчет по лабораторной работе с устной защитой, устный опрос. Контрольная работа.
3.8	Средства создания приложений с графическим интерфейсом пользователя	2			2			Отчет по лабораторной работе с устной защитой, устный опрос. Контрольная работа.
3.9	Вычисления в MATLAB. Обработка данных	2					2	Устный опрос, Отчет по лабораторной работе с устной защитой.

ИНФОРМАЦИОННО-МЕТОДИЧЕСКАЯ ЧАСТЬ

Перечень основной литературы

1. *Иванов, О. А.* Дискретная математика и программирование в Wolfram Mathematica: для бакалавров: учебное пособие для студентов направлений подготовки 01.03.02 "Прикладная математика и информатика", 01.03.04 "Прикладная математика", а также для студентов других направлений, изучающих дисциплину "Дискретная математика" / О. А. Иванов, Г. М. Фридман. - Санкт-Петербург [и др.]: Питер, 2020. - 349 с.
2. *Голубева, Л. Л.* Компьютерная математика. Символьный пакет Mathematica [Электронный ресурс] : электронный учебно-методический комплекс для специальности: 1-31 03 01 «Математика (по направлениям)», направление специальности: 1-31 03 01-02 «Математика (научно-педагогическая деятельность)» / Л. Л. Голубева [и др.] ; БГУ, Механико-математический фак., Каф. дифференциальных уравнений и системного анализа. – Минск: БГУ, 2021. Ссылка на ресурс: <https://elib.bsu.by/handle/123456789/276777>.
3. *Голубева, Л. Л.* Компьютерная математика. Символьный пакет MATHEMATICA : лабораторный практикум для студ. мех.-мат. фак. : в 2 ч. Ч. 1 / Л. Л. Голубева, А. Э. Малевич, Н. Л. Щеглова. - Минск : БГУ, 2012. - 235 с. - URL: <http://elib.bsu.by/handle/123456789/95686>.
4. *Чичурин, А. В.* Применение системы MATHEMATICA при решении дифференциальных уравнений и в задачах математического моделирования : курс лекций для студ. спец. 1-31 03 01 "Математика (по направлениям)" : в 3 ч. / А. В. Чичурин, Е. Н. Швычкина ; БГУ, Механико-математ. факультет. – Минск : БГУ, 2016 – 2017.
5. *Голубева, Л. Л.* Компьютерная математика. Числовой пакет MATLAB [Электронный ресурс] : электронный учебно-методический комплекс для специальности: 1-31 03 01 «Математика (по направлениям)», направление специальности: 1-31 03 01-02 «Математика (научно-педагогическая деятельность)» / Л. Л. Голубева [и др.] ; БГУ, Механико-математический фак., Каф. дифференциальных уравнений и системного анализа. – Минск: БГУ, 2021. Ссылка на ресурс: <https://elib.bsu.by/handle/123456789/276778>.

Перечень дополнительной литературы

1. *Голубева, Л. Л.* Компьютерная математика. Символьный пакет Mathematica: курс лекций / Л. Л. Голубева, А. Э. Малевич, Н. Л. Щеглова. Минск: БГУ, 2005. 103 с.
2. *Вавилов Н.А.* Mathematica для нематематика: Учеб. пособие для ВУЗов. / Н.А. Вавилов, В.Г. Халин, А.В. Юрков. М: Московский центр непрерывного математического образования, 2021. 483 с.

3. Седов, Е. С. Основы работы в системе компьютерной алгебры *Mathematica* : учебное пособие / Е. С. Седов. — 2-е изд. — Москва : ИНТУИТ, 2016. — 401 с. — Текст : электронный // Лань : электронно-библиотечная система. — URL: <https://e.lanbook.com/book/100339>
4. Воробьев, Е. М. Введение в систему символьных, графических и численных вычислений "Математика - 5": учеб. пособие для студ. вузов / Е.М. Воробьев. – Москва : Диалог-Мифи, 2005 – 365с. : ил. ; 20x14 см. – ISBN 5-86404-199-8.
5. Никулин Е.А. Компьютерная геометрия и алгоритмы машинной графики. СПб, БХВ – Петербург, 2015.
6. Gräbe, H.-G. *Mathematica* 6, Bafög-Ausgabe. Einführung, Grundlagen, Beispiele / H.-G. Gräbe, M. Kofler. München, Pearson Studium, 2007. 496 p.
7. Shifrin, L. *Mathematica* Programming: An Advanced Introduction. / L. Shifrin. Medium: e-book, 2008, 408 p.
8. Maeder, R. Computer Science with *Mathematica*: Theory and Practice for Science, Mathematics, and Engineering / R. Maeder. Cambridge Univ Pr, 2006. 389 p.
9. Lynch, S. Dynamical Systems with Application using *Mathematica* / S. Lynch. Birkhäuser Boston, 2007. 499 p.
10. Wickham-Jones, T. *Mathematica* Graphics. Techniques & Applications / T. Wickham-Jones. Springer Verlag New York Berlin Heidelberg, 1994, XIV, 721 p.
11. Sal Mangano. *Mathematica* Cookbook / Sal Mangano. O'Reilly Media, 2010. 832 p.
12. Wolfram, S. *Mathematica* book: 5 ed. / S. Wolfram. Wolfram, 2003. 1301 p.
13. Gilat, A. MATLAB: An Introduction with Applications / A. Gilat. — 6-th Edition. — Wiley, 2017. — 404 p
14. Лутц, М. Изучаем Python, том 1, 5-е изд.: Пер. с англ. / М. Лутц. СПб.: ООО “Диалектика”, 2019. 832 с.
15. Лутц, М. Изучаем Python, том 2, 5-е изд.: Пер. с англ. / М. Лутц. СПб.: ООО “Диалектика”, 2020. 720 с.
16. MATLAB Documentation [Electronic resource]. – The MathWorks, Inc, 2023. – Mode of access: <https://www.mathworks.com/help/matlab/>. – Date of access: 01.06.2023.
17. PDF Documentation for MATLAB [Electronic resource]. – The MathWorks, Inc, 2023. – Mode of access: https://www.mathworks.com/help/pdf_doc/matlab/ – Date of access: 01.06.2023.
18. Петряева, М. В. Применение MATLAB для решения аналитических задач моделирования: учебное пособие / М. В. Петряева, А. Н. Целых.

- Южный федеральный университет. – Ростов-на-Дону; Таганрог: Издательство Южного федерального университета, 2022. 131 с.
19. *Huei-Huang Lee*. Programming and Engineering Computing with MATLAB 2023 / Huei-Huang Lee. – SDC Publications, 2023. 530 p.
 20. *Oberbroeckling, L. A.* Programming Mathematics using MATLAB / L. A. Oberbroeckling. — 1st Edition. — Elsevier Academic Press, 2021. — 296 p.
 21. *Rosenbaum, D. A.* MATLAB Blues: How Behavioral Scientists and Others Can Learn From Mistakes for Better, Happier Programming / D.A. Rosenbaum. — 1st Edition. — Routledge, 2020. — 272 p.
 22. *Attaway, S.* MATLAB: a practical introduction to programming and problem solving / S. Attaway. — 5th Edition. — Elsevier Butterworth Heinemann, 2019. — 626 p.
 23. *Eshkabilov, S.* Beginning MATLAB and Simulink: From Novice to Professional / S. Eshkabilov. — Apress, 2019. — 544 p.
 24. *Голубева, Л. Л.* Компьютерная математика. Числовой пакет MATLAB: курс лекций / Л. Л. Голубева, А. Э. Малевич, Н. Л. Щеглова. Минск: БГУ, 2007. 164 с.
 25. *Голубева, Л. Л.* Компьютерная математика. Числовой пакет MATLAB: лабораторный практикум / Л. Л. Голубева, А. Э. Малевич, Н. Л. Щеглова. Минск: БГУ, 2008. 171 с.
 26. *Angermann, A.* MATLAB – Simulink – Stateflow. Grundlagen, Toolboxen, Beispiele / A. Angermann, M. Beuschel, M. Rau, U. Wohlfarth. Oldenburg Verlag München Wien, 2007. 495 p.
 27. *Adam, S.* MATLAB und Mathematik kompetent einsetzen / S. Adam. WILEY-VCH Verlag GmbH & Co. KGaA, Weinheim, 2006. 462 p.
 28. *Hanselman, D.* Mastering MATLAB / D. Hanselman, B. Littlefield. Pearson Educational, Inc, 2005. 852 p.
 29. *Свейгарт, Эл.* Python. Чистый код для продолжающих / Э. Свейгарт; [пер. с англ. Е. Матвеев]. Санкт-Петербург; Москва; Минск: Питер, 2022. – 383 с.
 30. *Копец, Дэвид.* Классические задачи Computer Science на языке Python = Classic Computer Science Problems in Python / Дэвид Копец ; [пер. с англ. Е. Полонской]. - Санкт-Петербург [и др.] : Питер, 2021. - 252 с.
 31. *Марк Саммерфилд,* Программирование на Python 3. Подробное руководство: Символ-Плюс, 2009, 608 с.
 32. *Уэс Маккинни,* Python и анализ данных: ДМК Пресс, 2015, 482 с.
 33. *Лутц, М.* Программирование на Python, том 1, 4-е изд.: Пер. с англ. / М. Лутц. СПб.: Символ-Плюс, 2011. 992 с.
 34. *Лутц, М.* Программирование на Python, том 2, 4-е изд.: Пер. с англ. / М. Лутц. СПб.: Символ-Плюс, 2011. 992 с.

Рекомендуемое учебно-лабораторное оборудование

Для проведения занятий требуются следующее программное обеспечение: операционная система MS Windows, MS Office, *Mathematica*, MATLAB, Python, JupyterLab.

Перечень рекомендуемых средств диагностики и методика формирования итоговой отметки

Согласно Положения о рейтинговой системе оценки знаний обучающихся по учебной дисциплине в БГУ контроль знаний студентов по дисциплине «Компьютерная математика» происходит в форме текущего контроля и промежуточной аттестации.

Объектом диагностики компетенций студентов являются знания, умения, полученные ими в результате изучения учебной дисциплины. Выявление учебных достижений студентов осуществляется с помощью мероприятий текущего контроля и промежуточной аттестации.

Для диагностики компетенций могут использоваться следующие средства текущего контроля: отчет по лабораторной работе с устной защитой, отчет по заданиям с устной защитой, устный опрос, контрольный опрос, контрольная работа.

Оценка текущего контроля знаний студента по дисциплине «Компьютерная математика» формируется в результате регулярной и систематической проверки знаний студентов во время занятий и по итогам их самостоятельной работы. Текущий контроль знаний проходит в форме опроса на лекциях и лабораторных занятиях, во время устной защиты отчета по лабораторным работам, отчета по заданиям с устной защитой, выполняемым в учебной лаборатории и самостоятельно вне аудитории, контрольных опросов и контрольных работ. Задания к лабораторным, контрольным работам, к отчетам с устной защитой составляются согласно содержанию учебного материала.

Во время самостоятельной работы студент выполняет задания, полученные на лабораторных занятиях, а также изучает рекомендуемую литературу. При защите лабораторных работ оценивается полнота ответа, аргументация выбранных решений, последовательность и оригинальность изложения материала, оригинальность кода, корректность оформления, самостоятельность выполнения заданий. Для совершенствования способностей учиться самостоятельно студентам могут выдаваться темы докладов, с которыми они выступают на занятиях.

Формой промежуточной аттестации по дисциплине «Компьютерная математика» учебным планом предусмотрен **зачет и экзамен**.

Зачет по дисциплине проходит в форме контрольного опроса в устной или письменной форме, выполнения заданий на компьютере. Если студент успешно защитил все лабораторные работы и получил отметку 4 и выше по контрольным работам, то допускается определение результатов промежуточной аттестации по дисциплине на основании результатов текущего контроля знаний без проведения дополнительного опроса на зачете. Это решение находится в компетенции преподавателя (группы преподавателей), ответственного за реализацию дисциплины. При этом явка студента на зачет является обязательной.

При формировании итоговой отметки используется рейтинговая система оценки знаний студента, дающая возможность проследить и оценить динамику процесса достижения целей обучения. Рейтинговая система предусматривает использование весовых коэффициентов в ходе проведения контрольных мероприятий текущей аттестации.

Примерные весовые коэффициенты, определяющие вклад текущей аттестации в отметку при прохождении промежуточной аттестации:

Формирование отметки за текущую аттестацию:

- отчеты по лабораторным работам с устной защитой, устный опрос, отчеты по заданиям с устной защитой – 60%;
- контрольные опросы и контрольные работы – 40%.

Итоговая отметка по дисциплине рассчитывается на основе отметки текущей аттестации (рейтинговой системы оценки знаний) – 40% и экзаменационной отметки – 60%.

Примерный перечень заданий для управляемой самостоятельной работы студентов

Тема 1.4. Функциональный стиль программирования (2 часа)

Примерный перечень заданий:

Задание. Построить функцию пользователя $Dif[expr, x]$, вычисляющую производную функции одной переменной.

- 1) Изучить встроенные функции дифференцирования, особенности использования каждой.
- 2) Проектировать операторы дифференциального и интегрального исчисления функций одной переменной.
- 3) Создать глобальные определения для дифференцирования функции одной переменной: правила дифференцирования, производные элементарных функций, производная сложной функции.
- 4) Тестировать созданные функции пользователя.

Целью данного задания является выработка у студента навыков функционального стиля программирования на языке Wolfram Language.

Форма контроля – отчет по заданиям с устной защитой

Тема 1.8. Построение системы «Аналитическая геометрия на плоскости» (2 часа)

Примерный перечень заданий:

Задание. Дополнить функционал системы «Аналитическая геометрия на плоскости» для решения задач компьютерной геометрии на плоскости.

Выполнить следующие задачи:

1. Спроектировать
 - объект «отрезок на плоскости», его конструкторы;
 - функции, вычисляющие свойства объектов, отношения между ними;
 - дополнительные конструкторы, запросы свойств для существующих объектов «точка на плоскости» и «прямая на плоскости»;
2. Спроектировать
 - объект «полигон на плоскости», его конструкторы;
 - функции, вычисляющие свойства полигона: направление обхода, периметр, площадь, выпуклость;
3. Написать спецификации созданных функций: имя, входные аргументы, что вычисляет, в каком виде возвращает результат вычислений. Алгоритмы описывать не нужно.
4. Разработать тестовые задачи для проверки работы созданных функций и тестировать их.

Целью данного задания является выработка у студента навыков объектно-ориентированного проектирования, функционального стиля программирования на языке Wolfram Language.

Форма контроля – отчет по заданиям с устной защитой

Тема 1.12. Принципы локализации переменных. Аппроксимация Паде (2 часа)

Примерный перечень заданий:

Задание. Изучить функцию, заданную параметрическими уравнениями координат $(x(t), y(t))$, непрерывно дифференцируемую n раз в окрестности точки (x_0, y_0) , приближая ее в окрестности этой точки многочленом Тейлора и соответствующей ему аппроксимацией Паде.

Создать функции пользователя, которые

- 1) вычисляют производную порядка n функции, заданной параметрически. Использовать рекурсивный способ задания;
- 2) строят многочлен Тейлора заданной функции в окрестности указанной точки;
- 3) строят аппроксимацию Паде, соответствующую вычисленному многочлену Тейлора;
- 4) отображают в графической области заданную функцию, многочлен Тейлора и аппроксимацию Паде.

Целью данного задания является выработка у студента навыков функционального стиля программирования на языке Wolfram Language, а также навыков исследования математических объектов.

Форма контроля – отчет по заданиям с устной защитой.

Тема 2.2. Объект как фундаментальное понятие в Python. Встроенные типы объектов и операции над ними (2 часа)

Примерный перечень заданий:

Задание 1. Выполните упражнения 1 – 11, представленные в [Лутц, М. Изучаем Python, том 1, 5-е изд.: Пер. с англ. / М. Лутц. СПб.: ООО “Диалектика”, 2019. 832 с.; с. 340 – 342]. Содержимое структурируйте. Выполнение сопровождайте краткими комментариями, располагая их или в ячейках кода, или в ячейках Markdown.

Задание 2. Проверьте приобретенные знания при изучении, ответив на контрольные вопросы, которые находятся в [Лутц, М. Изучаем Python, том 1, 5-е изд.: Пер. с англ. / М. Лутц. СПб.: ООО “Диалектика”, 2019. 832 с.; часть II], в конце каждой из глав 4 – 9.

Задания выполняются на основе методических указаний к лабораторной работе.

Целью УСР является изучение студентом встроенных типов объектов в Python, способов их создания и операций над ними; укрепление навыков структурирования интерактивных электронных блокнотов в JupyterLab.

Форма контроля – контрольная работа.

Тема 2.6. Функциональное программирование (2 часа)

Примерный перечень заданий:

Задание 1. Напишите пользовательскую функцию `func1` с произвольным количеством ключевых аргументов одного типа (числовые типы, строки, кортежи, списки), которая возвращает результат применения операции сложения ко всем аргументам функции.

Задание 2. Напишите генераторную функцию `func2` для генерации членов геометрической прогрессии $b_1 = 3$, $b_{n+1} = 5 b_n$, $n=1, 2, \dots$. На основе генераторной функции создайте кортеж из 10 первых членов геометрической прогрессии. Вычислите 10-ый член геометрической прогрессии с помощью функции `reduce` из модуля `functools`.

Задание 3. Напишите пользовательскую функцию `func3` с произвольным количеством позиционных и ключевых аргументов. Функция `func3` должна возвращать кортеж, состоящий из трех элементов. Первый элемент кортежа содержит число, соответствующее количеству вызовов пользовательской функции `func3`. Второй элемент кортежа содержит количество позиционных аргументов при вызове функции. Третий элемент кортежа содержит список ключей для ключевых аргументов при вызове функции `func3`. Напишите две реализации счетчика вызовов

пользовательской функции `func3`: с использованием глобальной переменной и с использованием синтаксиса функций-замыканий.

Задания выполняются на основе методических указаний к управляемой самостоятельной работе.

Целью УСР является выработка у студента навыков проектирования и написания функций, функционального стиля программирования на Python.

Форма контроля – контрольная работа.

Тема 2.8. Основы объектно-ориентированного программирования (2 часа)

Примерный перечень заданий:

Задание 1. Определите класс `Complex`, экземплярами которого являются комплексные числа. Определение класса должно содержать: метод инициализации экземпляра класса по действительной и мнимой части комплексного числа со значениями по умолчанию для каждого аргумента; метод, который для экземпляра класса `Complex` возвращает комплексно-сопряженное число в виде нового экземпляра класса `Complex`; метод `abs`, который возвращает модуль комплексного числа. Создайте один экземпляр класса `Complex` и осуществите вызов каждого метода.

Задание 2. Для работы со структурой данных стек определите класс `Stack` на основе встроенного класса `list` с одним методом вставки элемента в стек. Создайте экземпляр класса `Stack` и осуществите вставку элементов, генерируемых функцией `range(100)`.

Задание 3. Изобразите бинарное дерево поиска, соответствующее последовательной вставке в дерево элементов списка `[1, 2, 3, 1, 2, 3, 1, 2, 3]`. Напишите определение классов `BinaryTree`, `BinaryNode`, `EmptyNode` только с использованием методов инициализации. Поясните назначение каждого атрибута, определенного внутри методов инициализации. Поясните утверждение, что классы `BinaryTree`, `BinaryNode`, `EmptyNode` связаны на основе композиции. Можно ли в бинарное дерево поиска, созданное на основе элементов списка `[1, 2, 3, 1, 2, 3, 1, 2, 3]` с помощью классов `BinaryTree`, `BinaryNode`, `EmptyNode`, добавить элементы со значениями `True` или `False`? Ответ поясните.

Задание 4. Определите класс, который содержит атрибут класса, метод перегрузки операции, статический метод и классовый метод. Осуществите доступ к атрибуту класса и вызов каждого метода.

Задания выполняются на основе методических указаний к управляемой самостоятельной работе.

Целью УСР является выработка у студента навыков объектно-ориентированного проектирования, объектно-ориентированного стиля программирования на Python.

Форма контроля – контрольная работа.

Тема 3.5. Основы объектно-ориентированного программирования, классы и объекты (2 часа)

Примерный перечень заданий:

Задание 1. Переопределение встроенных функций и операторов для классов пользователя.

Задание 2. Реализация уникальных методов класса пользователя.

Задания выполняются на основе методических указаний к лабораторной работе.

Форма контроля – отчет по заданиям с устной защитой.

Тема 3.9. Вычисления в MATLAB. Обработка данных (2 часа)

Примерный перечень заданий:

Задание 1. *Исследование функций.* Нахождение нулей функции $y=f(x)$. Нахождение нулей функции, заданной полиномом. Поиск локальных минимумов функций одной и двух переменных.

Задание 2. *Задачи линейной алгебры.* Представление систем линейных уравнений в векторно-матричном виде. Решение СЛУ различными методами. Метод Крамера решения СЛУ.

Задание 3. *Интерполяция и аппроксимация данных.* Интерполяция одномерных данных полиномиальной функцией по методу наименьших квадратов. Сплайновая интерполяция функциями одной и двух переменных.

Задания выполняются на основе материалов лекции по теме.

Форма контроля – отчет по заданиям с устной защитой.

Описание инновационных подходов и методов к преподаванию учебной дисциплины

При организации образовательного процесса используется *эвристический подход*, который предполагает демонстрацию многообразия решений большинства профессиональных задач и жизненных проблем.

При организации образовательного процесса используется *практико-ориентированный подход*, который предполагает освоение содержания через решения практических задач.

При организации образовательного процесса *используются методы и приемы развития критического мышления*, которые представляют собой систему, формирующую навыки работы с информацией в процессе чтения и письма; понимании информации как отправного, а не конечного пункта критического мышления.

На лекциях и лабораторных занятиях используются следующие методы обучения: проблемного изложения, поисковый, репродуктивный, исследовательский. При проведении занятий также планируется

использовать наглядные методы, такие как иллюстрация, демонстрация, визуализация.

Методические рекомендации по организации самостоятельной работы обучающихся

Для организации самостоятельной работы студентов по учебной дисциплине рекомендовано разместить на образовательном портале или сайте кафедры учебно-методические материалы: курсы лекций и лабораторные практикумы, методические указания к лабораторным занятиям, вопросы для подготовки к зачету и экзамену, перечень рекомендуемой литературы, информационные ресурсы.

Самостоятельная работа студента включает в себя работу с учебной литературой по заданным разделам дисциплины, поиск в Интернете новейшей учебной и научной информации в указанных областях знаний и знакомство с ней, а также выполнение задач, поставленных на занятиях.

Примерный перечень вопросов к экзамену

Раздел 1. Символьный пакет *Mathematica*

1. Составные части *Mathematica*, назначение каждой. Приобретение знаний *Mathematica* во время сеанса работы *Session*.
2. Особенности интерфейса пользователя *Mathematica*. Структура системы справки. Сообщения и предупреждения.
3. Рабочий документ *Mathematica*: структура, форматирование, редактирование. Служебные и горячие клавиши. Особенности ввода: типы ячеек, математические выражения, специальные символы.
4. Главный цикл *the Main Loop* обработки выражения `In[] – Out[]`.
5. Рекуррентное определение выражения. Типы атомарных объектов. Извлечение части атомарного выражения.
6. Полная, или каноническая форма записи выражения. Функции, отображающие структуру выражения.
7. Структура выражения. Определения понятий уровня выражения, глубины выражения, позиции подвыражения.
8. Функции `Level`, `Depth`, `Position`. Спецификатор уровней выражения *levelspec*, его использование.
9. Функции для извлечения частей выражения, не являющегося атомарным: `Extract`, `Take`, `Part`, `Select`, `Cases`, `DeleteCases`.
10. Списки как основное используемое выражение. Создание списка, изменение его структуры, порядка следования элементов, добавление и удаление элементов списка.

11. Функция без имени: определение, построение, использование. Формы записи безымянной функции. Понятие выражения-однострочника, правила его написания.
12. Образец `Pattern`: определение, назначение, способы именования. Функция для сравнения выражения с образцом.
13. Виды образцов: с условием, содержащие альтернативу, повторяющиеся, с заданными по умолчанию значениями, со встроенными по умолчанию значениями.
14. Использование образцов в аргументах встроенных функций *Mathematica* и в функциях пользователя.
15. Базовые функции двумерной и трехмерной графики.
16. Двумерные графические примитивы. Директивы. Опции.
17. Основные трехмерные графические примитивы. Директивы. Опции.
18. Правила преобразований: определение, классификация, сфера видимости и момент времени вычисления. Использование правил.
19. Пополнение списков значений символа при опеределении глобальных правил преобразований. Место глобального правила в списке.
20. Локальные правила преобразований, или подстановки.
21. Функции для выполнения локальных правил преобразований. Программирование в стиле локальных правил.
22. Функции семейства `Map`: назначение, аргументы, общие принципы работы. Особенности использования каждой функции. Назначение и использование функции `Apply`.
23. Функции `Map`, `MapAt`, `MapAll`, `MapThread`, `MapIndexed`: назначение, входы и выходы. Примеры.
24. Итерационные процессы. Операторы семейств `Nest` и `Fold`. Их назначение, общее и различие между этими группами.
25. Структуры выбора и структуры повторения, их реализация в *Mathematica*.
26. Локализация сферы видимости переменных. Механизмы локализации символов в структурах `Block`, `Module`, `With`, `Function`.
27. Стандартный порядок вычисления выражения ядром. Изменение стандартного порядка вычисления выражения при наличии атрибутов семейства `xxHoldxxx`.
28. Атрибуты: назначение, установка, момент вычисления на примере установленных свойств символа `Plus`. Особенности вычисления символа, который наделен атрибутом семейства `Hold`.

29. Динамическая визуализация выражений. Функции `Manipulate` и `Dynamic`.
30. Динамическая визуализация графических объектов. Элементы управления, их использование.

Раздел 2. Компьютерное моделирование на Python

1. Свойства и особенности языка программирования Python. Стандартная библиотека Python. Концептуальная структура программы. Файловая структура программы.
2. Python как интерпретируемый язык программирования. Трансляция в байт-код. Виртуальная машина Python. Расширения Python.
3. Объект. Встроенные типы объектов. Литералы и конструкторы для создания объектов встроенных типов. Изменяемые и неизменяемые типы. Булево значение объекта.
4. Последовательность. Встроенные типы объектов последовательностей. Операции над последовательностями: индексация, конкатенация, повторение, длина, проверка принадлежности.
5. Словарь. Создание словаря. Операции над словарем. Методы словаря.
6. Списковое включение и его аналоги. Списковое включение с конструкцией выбора. Списковые включения с несколькими уровнями вложенности. Примеры.
7. Файловый объект. Создание файлового объекта. Методы файловых объектов. Итерирование файлового объекта.
8. Строки документации и комментарии. Правила оформления строк документации. Атрибут `__doc__`. Инструмент `PyDoc`.
9. Синтаксический шаблон составного оператора. Оператор выбора `if`. Операторы цикла `for` и `while`.
10. Переменная, ее создание, использование и удаление. Оператор присваивания и его формы. Неявные присваивания. Пространства имен.
11. Пользовательская функция. Оператор определения функции `def`. Определение функции со значениями по умолчанию для аргументов. Определение функции с произвольным количеством аргументов. Объект функции. Аннотация типов при определении функции.
12. Позиционные и ключевые аргументы при вызове функции. Позиционный и ключевой режимы сопоставления объектов-значений и аргументов-переменных при вызове функции.
13. Определение и применение `lambda`-функций. Примеры.

14. Переменная. Типы областей видимости переменной. Правило лексических областей при поиске переменной. Операторы объявления **global** и **nonlocal**.
15. Механизм итераций. Итерируемый объект и его элементы. Объект итератора.
16. Механизм итераций. Ручное и автоматическое выполнение итераций. Итерационные инструменты. Протокол итераций.
17. Генераторный объект. Генераторное выражение. Генераторная функция. Операторы **yield** и **yield from**. Преимущества генераторных объектов перед коллекциями.
18. Функции **map** и **filter** для организации циклов на итерируемых объектах. Примеры.
19. Спецификации функций **enumerate** и **zip**. Использование функций **enumerate** и **zip** в циклах **for**. Использование функции **zip** для работы со словарями.
20. Функция-замыкание. Фабричная функция. Пример определения функции-замыкания со счетчиком вызовов функции.
21. Модуль. Пакет. Операторы импортирования модуля **import** и **from**. Загрузка и перезагрузка модуля.
22. Модуль. Глобальная область видимости переменной. Объект модуля. Пространство имен объекта модуля. Атрибут **__name__** объекта модуля.
23. Оператор создания класса **class**. Объект класса. Экземпляр класса. Атрибуты и методы класса. Объект связанного метода. Концепция инкапсуляции. Метод инициализации экземпляра класса.
24. Дерево наследования. Суперклассы. Подклассы. Множественное наследование. Настройка кода за счет замены метода. Настройка кода за счет расширения кода метода.
25. Перегрузка операций и встроенных функций. Методы строкового представления. Метод вызова. Методы операции сложения.
26. Перегрузка операций и встроенных функций. Методы операций сравнения. Методы булевого значения. Методы операций индексации.
27. Исключение. Операторы для работы с исключениями. Стандартный обработчик исключений. Встроенные исключения. Пользовательские исключения. Диспетчер контекста.
28. Исключение. Оператор **try** для перехвата и обработки исключений. Операторы **raise** и **assert** для генерации исключений.
29. Расширение **numpy**. Тип данных **ndarray**. Атрибуты массива. Итерирование массивов. Создание массивов. Индексация массивов.

30. Расширение **numpy**. Методы для изменения структуры массива. Атрибут **shape**. Представление и копия массива. Атрибут **base**.
31. Расширение **numpy**. Векторизация вычислений. Представление вектора и матрицы с помощью массива. Матричное произведение. Скалярное произведение. Транспонирование матрицы.
32. Пакет **matplotlib**. Базовые функции двумерной графики. Функции для создания и оформления текстовых объектов в графической области.
33. Пакет **matplotlib**. Графические объекты для представления изображения. Иерархическая структура изображения. Объектно-ориентированный подход для построения изображения.

Раздел 3. Числовой пакет MATLAB

1. Назначение, структура пакета MATLAB, сравнение с другими математическими пакетами.
2. Интерфейс системы MATLAB. Командное окно. Рабочее пространство. История команд. Текущий каталог. Редактор-отладчик файлов, редактор Live Editor. Графические окна. Работа в интерактивном режиме.
3. Встроенные типы данных. Иерархия типов данных. Векторы, матрицы, тензоры.
4. Встроенные типы данных. Операции с числовыми данными. Форматы представления числовых данных.
5. Встроенные типы данных. Символьные и текстовые массивы.
6. Встроенные типы данных. Структуры. Массивы ячеек. Таблицы. Категориальные данные.
7. Операторы управления.
8. Программирование М-сценариев и М-функций. Функции, определяемые пользователем. Структура типичной функции в MATLAB. Входные и выходные аргументы функции. Локальные и глобальные переменные.
9. Типы функций. Первичные функции, подфункции, частные функции. Анонимные функции. Перегрузка функций.
10. Классы пользователя. Основные принципы ООП, поддерживаемые в MATLAB. Классы и объекты. Классы, определяемые пользователем.
11. Классы пользователя. Типы конструкторов классов. Свойства и методы классов. Основные методы канонических классов MATLAB. Способы создания классов пользователей в MATLAB.
12. Ввод-вывод данных. Типы форматов файлов данных, поддерживаемые MATLAB.

13. Ввод-вывод данных. Экспорт данных в MAT-файлы. Импорт данных из MAT-файлов. Бинарные данные. ASCII (текстовые) данные.
14. Ввод-вывод данных. Чтение форматированных данных из файлов. Запись форматированных данных в файл.
15. Графика. Основы высокоуровневой и дескрипторной графики. Построение двумерных и трехмерных графических объектов. Построение графиков функций в одной системе координат, в разных с/к, в разных графических окнах.
16. Графика. Иерархия классов графических объектов. Доступ к свойствам графических объектов.
17. Графика. Управление камерой. Точка обзора. Азимут и возвышение. Позиционирование камеры для просмотра графического объекта или группы объектов.
18. Элементы дескрипторной графики для создания графического интерфейса пользователя GUI. Программирование событий callbacks GUI-компонентов. Установка свойств компонентов в Инспекторе свойств объекта.
19. Создание GUI в среде разработки приложений GUIDE. Инструмент App Designer.
20. Вычисления в MATLAB. Передача указателя на функцию в качестве аргумента другой функции.
21. Вычисления в MATLAB. Исследование функций: нахождение нулей функции, поиск локальных экстремумов функций одной и двух переменных.
22. Вычисления в MATLAB. Решение систем линейных уравнений в MATLAB.
23. Вычисления в MATLAB. Задачи интерполяции.

ПРОТОКОЛ СОГЛАСОВАНИЯ УЧЕБНОЙ ПРОГРАММЫ УВО

Название учебной дисциплины, с которой требуется согласование	Название кафедры	Предложения об изменениях в содержании учебной программы УВО по учебной дисциплине	Решение, принятое кафедрой, разработавшей учебную программу (с указанием даты и номера протокола) ²
Алгебра и теория чисел	Кафедра высшей алгебры и защиты информации	нет	Вносить изменения не требуется (протокол № 13 от 14.06.2023)
Математический анализ	Кафедра теории функций	нет	Вносить изменения не требуется (протокол № 13 от 14.06.2023)
Аналитическая геометрия	Кафедра геометрии, топологии и методики преподавания математики	нет	Вносить изменения не требуется (протокол № 13 от 14.06.2023)
Дифференциальные уравнения	Кафедра дифференциальных уравнений и системного анализа	нет	Вносить изменения не требуется (протокол № 13 от 14.06.2023)
Методы программирования	Кафедра веб-технологий и компьютерного моделирования	нет	Вносить изменения не требуется (протокол № 13 от 14.06.2023)

ДОПОЛНЕНИЯ И ИЗМЕНЕНИЯ
К УЧЕБНОЙ ПРОГРАММЕ ПО ИЗУЧАЕМОЙ УЧЕБНОЙ
ДИСЦИПЛИНЕ НА _____ / _____ УЧЕБНЫЙ ГОД

№ п/п	Дополнения и изменения	Основание

Учебная программа пересмотрена и одобрена на заседании кафедры
 _____ (протокол № _____ от _____ 200__ г.)
 (название кафедры)

Заведующая кафедрой
 кандидат физ.-мат. наук, доцент _____
 (ученая степень, ученое звание) (подпись)

Л.Л. Голубева
 (И.О.Фамилия)

УТВЕРЖДАЮ
 Декан факультета
 доктор физ.-мат. наук, профессор _____
 (ученая степень, ученое звание) (подпись)

С.М. Босяков
 (И.О.Фамилия)