

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ КЫРГЫЗСКОЙ
РЕСПУБЛИКИ
СОВРЕМЕННЫЙ МЕЖДУНАРОДНЫЙ УНИВЕРСИТЕТ**

КАФЕДРА ИНФОРМАТИ

«Согласовано»

«Утверждаю»

Начальник учебной части

_____Максат Макамбай

_____Ерке Е.Э.

«_____» _____20__г

«_____» _____20__г



**РАБОЧАЯ ПРОГРАММА
Практикум решения
математических задач на ЭВМ (MATLAB)**

для специальности (направления специальности): **510200 –**

Прикладная математика и информатика

Форма обучения: очная

Виды учебной деятельности:

Програм ма	Всего часов				СР С	СРС П	Форма отчетно сти
	всег о	аудиторн ых	лекц ия	Прак.р аб			
2-курс, 4 семестр	4 кр 120 час	60 час		60 ч	20 ч	40 ч	экзамен

«Согласовано»

«Рассмотрено»

Профилирующей кафедрой

на заседании

кафедры _____

_____зав.каф.

от «_____» _____20__г.

Рабочая программа дисциплины разработана в соответствии с ГОС (1578/1 от 21 сентября 2021 г), ОПОП направления **510200 – Прикладная математика и информатика**. Положения о разработке УМК в СМУ.

Разработчик: _____Э.К. Эркулова

Жалал-Абад

2023 г.

Содержание

Силлабус	ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.
1.1 реквизиты дисциплины	3
реквизиты преподавателя	ОШИБКА! ЗАКЛАДКА НЕ ОПРЕДЕЛЕНА.
1.2. Объем дисциплины и виды учебной работы	3
2. Место дисциплины в структуре ооп	3
3. Пререквизиты и постреквизиты курса	4
пререквизиты:	4
основы математики: студенты должны иметь базовые знания в области алгебры, тригонометрии, дифференциального и интегрального исчисления.	4
опыт программирования: желательно иметь некоторый опыт программирования, хотя бы на базовом уровне. Знание основных концепций программирования облегчит изучение матлаба.	4
основы работы с эвм: студентам рекомендуется иметь базовые навыки работы с компьютером и операционной системой, а также понимание основных понятий информатики.	4
постреквизиты:	4
углубленное понимание математических концепций: после завершения курса студенты должны иметь углубленное понимание математических методов, используемых в матлабе, и умение применять их для решения различных задач.	4
навыки программирования на матлабе: после завершения курса студенты должны иметь навыки программирования на матлабе, включая работу с матрицами, построение графиков, написание скриптов и функций.	4
умение анализировать и интерпретировать результаты: студенты должны уметь анализировать результаты, полученные при помощи матлаба, и интерпретировать их с точки зрения математической модели или проблемы, которую они решают.	4
4. Необходимость изучения курса	4
6 компетенции обучающегося, формируемые в результате освоения дисциплины	5
6 образовательные технологии	6
8 критерий оценки знаний студентов на экзамене	7
8.1. Оценка знаний (академической успеваемости) осуществляется по 100 балльной системе (шкале) следующим образом:	9
8.2. Технологическая карта дисциплины	9
3. Тематический план и содержание учебной дисциплины	9
« практикум решения математических задач на эвм »	9
срсп	10
срс	12
10. Учебно-методическое, информационное и материально-техническое обеспечение дисциплины	14
11. Вопросы для подготовки к экзамену	14
1. Методические указания для обучающихся по освоению дисциплины (модуля)	23

20.			
21.			
22.			
23.			
24.			
25.			

1.1 Реквизиты дисциплины

Дисциплина : «Практикум решения математических задач на ЭВМ»

Курс: 3

Направление: 510200 – Прикладная математика и информатика

Количество кредит часов: 4 кр (120 ч)

Форма обучения: очная

1.2. Объем дисциплины и виды учебной работы

Вид учебной работы	Семестр 4
	Всего часов 120 ч
Аудиторные занятия (всего)	60
В том числе:	
Лекции (Л)	
Практические занятия (ПЗ)	60
Семинары (С)	
Лабораторные практикумы (ЛП)	
Клинические практические занятия (КПЗ)	
Самостоятельная работа (всего)	60
СРС	20
СРС	40
Форма контроля	экзамен
Общая трудоемкость (час.)	120 ч

2. Место дисциплины в структуре ООП

Учебная дисциплина «Финансовая математика» является специальной дисциплиной, формирующей профессиональные знания, необходимые для будущей трудовой деятельности.

Дисциплина относится к вариативной части профессионального цикла.

3. Пререквизиты и постреквизиты курса

Пререквизиты:

Основы математики: студенты должны иметь базовые знания в области алгебры, тригонометрии, дифференциального и интегрального исчисления.

Опыт программирования: желательно иметь некоторый опыт программирования, хотя бы на базовом уровне. Знание основных концепций программирования облегчит изучение Матлаба.

Основы работы с ЭВМ: студентам рекомендуется иметь базовые навыки работы с компьютером и операционной системой, а также понимание основных понятий информатики.

Постреквизиты:

Углубленное понимание математических концепций: после завершения курса студенты должны иметь углубленное понимание математических методов, используемых в Матлабе, и умение применять их для решения различных задач.

Навыки программирования на Матлабе: после завершения курса студенты должны иметь навыки программирования на Матлабе, включая работу с матрицами, построение графиков, написание скриптов и функций.

Умение анализировать и интерпретировать результаты: студенты должны уметь анализировать результаты, полученные при помощи Матлаба, и интерпретировать их с точки зрения математической модели или проблемы, которую они решают.

4. Необходимость изучения курса

1. Изучение курса "Практикум решения математических задач на ЭВМ (Матлаб)" полезно для:
- 2.
3. Повышения навыков программирования на MATLAB.
4. Применения в реальных проектах и исследованиях.
5. Улучшения аналитических способностей.
6. Подготовки к научной работе.
7. Увеличения шансов на рынке труда.

Таким образом, изучение курса "Практикум решения математических задач на ЭВМ (Матлаб)" может быть полезным для

График проведенных практических занятий

	Дата проведения лекционного занятия по расписанию	Дата проведения практического занятия по расписанию	Дата проведения по расписанию СРСР
1.			
2.			
3.			
4.			
5.			
6.			
7.			
8.			
9.			
10.			
11.			
12.			
13.			
14.			
15.			
16.			
17.			
18.			
19.			

вычисления). Размещайте их в LMS до аудиторных занятий, чтобы на практике студенты сразу писали код, а не слушали теорию.

– **Инструкции по виртуальным лабораториям.** Составьте подробные «how-to» документы для работы в облачном MATLAB, JupyterLab или с VR-модулями. Предусмотрите скриншоты, шаблоны проектов и чек-листы.

– **Геймификация.** Разработайте систему бейджей и уровней за успешное решение мини-челленджей (точность расчёта погрешностей, скорость выполнения оптимизации). Разместите рейтинги в цифровой доске достижений LMS.

3. Организация занятий и проектов

– **Практикум-хакатоны.** Проводите мини-хакатоны (24–48 часов), где студенты в командах решают сложные ОДУ или СЛАУ с параллельными вычислениями, GPU-ускорением и отладкой в реальном времени.

– **VR-кейсы.** Если есть доступ к VR-оборудованию или WebVR, создайте виртуальную «экскурсию» по этапам численного решения, где студенты «погружаются» в трёхмерную модель системы.

– **Cloud-проекты.** Используйте GitHub–Colab или MATLAB Online вместе с CI (GitHub Actions), чтобы при каждом коммите запускались автоматические тесты и генерировались отчёты.

4. Формы контроля и обратная связь

– **Learning analytics.** Настройте сбор метрик (время выполнения заданий, количество попыток, посещаемость модулей) и анализируйте результаты в Power BI. На основе дашборда корректируйте планы занятий.

– **Peer-review.** Организуйте в LMS обмен MATLAB-скриптами между студентами для взаимного кода-ревью: каждый студент получает два чужих проекта, оставляет комментарии и выставляет оценку по рубрике.

– **Инклюзивная оценка.** Предусмотрите доступные форматы отчётов: аудио-описания графиков, цвета-контрасты по WCAG, мобильные версии материалов для студентов с ОВЗ.

5. Профессиональное развитие преподавателя

– **Участие в сообществах.** Поддерживайте связь с коллегами на платформах MathWorks, участвуйте в вебинарах и конференциях по EdTech.

– **Рефлексия и оценка.** После каждого модуля собирайте анонимную обратную связь: что сработало, а что требовало бы доработки, и вносите изменения по циклу PDCA.

– **Документирование практик.** Ведите методический журнал: кратко описывайте проведённые инновационные мероприятия, их эффект на успеваемость и мотивацию студентов, сохраняйте примеры лучших проектов.

Следуя этим указаниям, преподаватель сможет не только обучить студентов традиционным численным методам, но и сформировать у них навыки самостоятельного освоения новых технологий, коллективного проектирования и критического анализа результатов.

студентов, желающих расширить свои знания и навыки в области программирования и математики, а также для тех, кто планирует связать свою карьеру с научными исследованиями или применением математических методов в практической деятельности.

5. Цели и задачи курса

Цель освоения дисциплины:

– дать студентам необходимые знания о разработке прикладных программ,

– методики постановки, подготовки и решения инженерных задач на современных персональных компьютерах;

– освоение современных методов программирования;

– овладение знаниями по основным разделам информатики.

Задачи курса:

– научить студентов эффективно использовать возможности среды программирования MatLab для решения стандартных задач;

– научить использовать различные виды алгоритмов при решении задач;

– научить использовать компьютер для решения инженерных задач;

– развить логическое и алгоритмическое мышление студентов.

6 Компетенции обучающегося, формируемые в результате освоения дисциплины

2. Код Компетенция Описание

ПК 2 Освоение новых знаний	Самостоятельно осваивает и применяет новые вычислительные пакеты (MATLAB, Python-библиотеки, облачные сервисы), AI-ассистентов и современные численные методы решения задач.
ПК 4 Организационно-управленческая деятельность	Планирует и координирует командную разработку вычислительных проектов с использованием Agile/Scrum-практик, распределяет задачи и следит за выполнением через цифровые трекеры.
ПК 10 Прикладное программирование	Разрабатывает, отлаживает и оптимизирует код для решения прикладных математических задач, используя виртуальные лаборатории,

		параллельные вычисления и GPU-ускорение.
ПК 14	Педагогическая деятельность	Владеет методикой проведения практикумов и консультаций по численным методам и программированию с применением flipped classroom, интерактивных симуляторов и вебинаров.
ПК 16	Социально-ориентированная деятельность	Реализует вычислительные решения для социальных и образовательных проектов (open data, e-learning), участвует в хакатонах и инициативах по цифровой грамотности населения.

В результате освоения дисциплины "Практикум решения математических задач на ЭВМ (Матлаб)" студенты формируют следующие компетенции:

- 1. Навыки программирования:** Студенты приобретают уверенность в использовании MATLAB для написания скриптов и функций, что развивает их программистские навыки.
- 2. Математическое моделирование:** Студенты учатся строить математические модели для решения различных задач и применять программные инструменты для их анализа и решения.
- 3. Аналитическое мышление:** Освоение курса требует от студентов умения анализировать сложные математические задачи, выделять основные аспекты и разрабатывать эффективные методы их решения.
- 4. Компетенции в области численных методов:** Студенты изучают различные численные методы, используемые для решения математических задач, и приобретают опыт их применения с использованием MATLAB.
- 5. Умение интерпретировать результаты:** Студенты учатся анализировать полученные результаты, оценивать их точность и правильность, а также делать выводы на основе этих результатов.
- 6. Креативное решение проблем:** Освоение курса требует от студентов способности к творческому подходу к решению задач и поиску альтернативных методов и подходов.

Эти компетенции являются важными как для академического, так и для профессионального развития студентов в области математики, программирования и научных исследований.

6 Образовательные технологии

- `SVMModel = fitcsvm(X,Y);`
- **Как использовать MATLAB для анализа временных рядов?** MATLAB для анализа временных рядов можно использовать с помощью функций для работы с временными рядами, статистических функций и др.
- **Как создать и использовать скользящее среднее в MATLAB?** Скользящее среднее в MATLAB можно создать и использовать с помощью функции `movmean()`. Например:

```
scss
smoothedData = movmean(data, windowSize);
```
- **Каким образом можно реализовать авторегрессионную модель в MATLAB?** Авторегрессионную модель в MATLAB можно реализовать с помощью функции `ar()`. Например:

```
scss
model = ar(data, order);
```
- **Как использовать волновой анализ в MATLAB?** Волновой анализ в MATLAB можно использовать с помощью функций `wavedec()` и `wrcoef()`. Например:

```
scss
[c,l] = wavedec(data, level);
```

1. Методические указания для преподавателя по освоению дисциплины «Практикум решения математических задач на ЭВМ» с инновационными компонентами

Данные рекомендации помогут эффективно внедрить в практику решения математических задач на ЭВМ современные образовательные технологии, сделать занятия более интерактивными и ориентированными на реальные задачи.

1. Проектирование курса с инновационными компонентами

- **Формулировка целей и результатов.** Помимо традиционных умений решать задачи в MATLAB, укажите, что студент приобретёт опыт работы с облачными средами (MATLAB Online), AI-ассистентами (ChatGPT-API), инструментами learning analytics и веб-технологиями для визуализации.
- **Структура дисциплины.** Разбейте программу на модули, в которых каждый блок содержит элемент «инновационной практики» (VR-кейс, геймификация, flipped-classroom или cloud-проект).

2. Подготовка и методическое обеспечение

- **Видеоуроки и flipped-classroom.** Запишите короткие видео о ключевых приёмах MATLAB (например, табулирование функций или параллельные

• **Каким образом можно выполнить фильтрацию изображения в MATLAB?** Фильтрацию изображения в MATLAB можно выполнить с помощью функций `imfilter()` или `medfilt2()`. Например:

```
scss
```

- `filteredImage = imfilter(image, filter);`

• **Как создать фильтр высоких частот для изображения в MATLAB?** Фильтр высоких частот для изображения в MATLAB можно создать с помощью функции `fspecial()`. Например:

```
scss
```

- `filter = fspecial('high');`

• **Как применить преобразование Фурье к изображению в MATLAB?** Преобразование Фурье к изображению в MATLAB можно применить с помощью функции `fft2()`. Например:

```
scss
```

- `F = fft2(image);`

• **Как выполнить сегментацию изображения в MATLAB?** Сегментацию изображения в MATLAB можно выполнить с помощью функций `imbinarize()`, `bwconncomp()` и др.

• **Каким образом можно использовать MATLAB для обработки данных?** MATLAB для обработки данных можно использовать с помощью функций для работы с таблицами, статистических функций и др.

• **Как создать и выполнить кластеризацию данных в MATLAB?** Кластеризацию данных в MATLAB можно создать и выполнить с помощью функций `kmeans()`, `clusterdata()` и др.

• **Как использовать метод главных компонент для анализа данных в MATLAB?** Метод главных компонент для анализа данных в MATLAB можно использовать с помощью функции `pca()`. Например:

```
scss
```

- `[coeff,score,latent,~,explained] = pca(data);`

• **Как создать и использовать нейронные сети в MATLAB?** Нейронные сети в MATLAB можно создать и использовать с помощью Neural Network Toolbox или функций для работы с нейронными сетями.

• **Каким образом можно реализовать метод опорных векторов в MATLAB?** Метод опорных векторов в MATLAB можно реализовать с помощью функции `fitcsvm()` или `svmtrain()`. Например:

```
scss
```

Изучение дисциплины предполагает использование традиционных способов коллективного обучения – лекций, лабораторных занятий, индивидуальных заданий с последующей отчетностью.

Применяемые информационные технологии: лекции в форме презентаций, обучающие и тестирующие программы, электронные учебники.

Оценочные средства для текущего контроля успеваемости, рубежного контроля по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы бакалавров:

- **формой текущего контроля знаний студентов (аудиторные занятия)** является контроль посещения лекционных и практических занятий, активность студентов на практических занятиях. Каждый из двух текущих контролей оценивается по **30 баллов**.

формой текущего контроля знаний студентов (внеаудиторные занятия) является контроль СРСР, участие в НИРС (выступление на студенческой конференции, публикация статей)

- **формой итогового контроля знаний и умений бакалавров по курсу** является экзамен.

Текущий и рубежный контроль. Студенты после выполнения соответствующих (первому или второму модулю) практических и лабораторных работ допускаются к рубежному контролю. Каждый из двух рубежных контролей (модулей) оценивается по **30** балльной шкале.

Итоговый контроль. Итоговый контроль реализуется в форме защиты собственно созданных программ (в виде компьютерного тестирования) и оценивается по **30** балльной шкале.

Правила оценивания рубежного и итогового контроля.

8 Критерий оценки знаний студентов на экзамене

Выставление оценок на экзаменах осуществляется на основе принципов объективности, справедливости, всестороннего анализа качества знаний студентов, и других положений, способствующих повышению надежности оценки знаний обучающихся, и устранению субъективных факторов.

В соответствии с действующими нормативными актами и рекомендациями Министерства образования и науки КР устанавливаются следующие критерии выставления оценок на экзаменах:

- **оценка "отлично"** выставляется студенту, который обнаружил на экзамене всестороннее, систематическое и глубокое знание учебно-программного материала, умение свободно выполнять задания, предусмотренные программой, который усвоил основную литературу и ознакомился с дополнительной литературой, рекомендованной программой. Как правило, оценка "отлично" выставляется студентам, усвоившим взаимосвязь основных понятий дисциплины и их значений для приобретаемой профессии, проявившим творческие способности в понимании, изложении и использовании учебно-программного материала;

- **оценка "хорошо"** выставляется студенту, который на экзамене обнаружил полное знание учебно-программного материала, успешно выполнил предусмотренные в программе задания, усвоил основную литературу, рекомендованную в программе. Как правило, оценка "хорошо" выставляется студентам, показавшим систематический характер знаний по дисциплине и способным к их самостоятельному выполнению и обновлению в ходе дальнейшей учебной работы и профессиональной деятельности;

- **оценка "удовлетворительно"** выставляется студенту, обнаружившему знание основного учебного материала в объеме, необходимом для дальнейшей учебы и предстоящей работы по профессии, справляющемуся с выполнением заданий, предусмотренных программой, который ознакомился с основной литературой, рекомендованной программой. Как правило, оценка "удовлетворительно" выставляется студентам, допустившим погрешности в ответе на экзамене и при выполнении экзаменационных заданий, но обладающим необходимыми знаниями для их устранения под руководством преподавателя;

- **оценка "неудовлетворительно"** выставляется студенту, обнаружившему пробелы в знаниях основного учебно-программного материала, допустившему принципиальные ошибки в выполнении предусмотренных программой заданий, не ознакомившемуся с основной литературой, предусмотренной программой, и не овладевшему базовыми знаниями, предусмотренными по данной дисциплине и определенными соответствующей программой курса (перечень основных знаний и умений, которыми должны овладеть студенты, является обязательным элементом рабочей программы курса).

Каким образом можно выполнить визуализацию данных в MATLAB? Визуализацию данных в MATLAB можно выполнить с помощью функций `plot()`, `scatter()`, `bar()` и др.

Как создать анимацию в MATLAB? Анимацию в MATLAB можно создать с помощью функции `movie()`, `set()` и `get()`. Например:

```
• for i = 1:10
    % ваш код обновления данных
    frame = getframe(gcf);
    movieVector(i) = frame;
end
```

Каким образом можно использовать MATLAB для обработки сигналов? MATLAB для обработки сигналов можно использовать с помощью Signal Processing Toolbox или функций для работы с сигналами.

Как реализовать фильтрацию сигнала в MATLAB? Фильтрацию сигнала в MATLAB можно реализовать с помощью функций `filter()`, `designfilt()` и др.

Как создать фильтр нижних частот в MATLAB? Фильтр нижних частот в MATLAB можно создать с помощью функции `designfilt()`. Например:

```
arduino
• d = designfilt('lowpassfir', 'FilterOrder',
10, 'CutoffFrequency', 0.5);
```

Как реализовать фурье-преобразование в MATLAB? Фурье-преобразование в MATLAB можно реализовать с помощью функции `fft()`. Например:

```
scss
• Y = fft(y);
```

Как построить график спектра сигнала в MATLAB? График спектра сигнала в MATLAB можно построить с помощью функции `plot()`. Например:

```
scss
• plot(frequency, abs(Y));
```

Как использовать MATLAB для обработки изображений? MATLAB для обработки изображений можно использовать с помощью Image Processing Toolbox или функций для работы с изображениями.

```
myTable = table(data1, data2, 'VariableNames', {'Var1', 'Var2'});
```

Как реализовать интерполяцию данных в MATLAB? Интерполяцию данных в MATLAB можно реализовать с помощью функции `interp1()`.

Например:

```
interpData = interp1(x, y, newX);
```

Каким образом можно генерировать случайные числа в MATLAB?

Случайные числа в MATLAB можно генерировать с помощью функций `rand()` и `randn()`. Например:

```
randomNumber = rand();
```

Как использовать символьные вычисления в MATLAB?

Символьные вычисления в MATLAB можно использовать с помощью Symbolic Math Toolbox или функции `sym()`. Например:

```
makefile
```

```
syms x;
```

```
f = x^2 + 2*x + 1;
```

Как создать и использовать символьные переменные в MATLAB?

Символьные переменные в MATLAB можно создать с помощью функции `sym()`. Например:

```
syms x y z;
```

Как решать символьные уравнения в MATLAB? Символьные уравнения в MATLAB можно решать с помощью функции `solve()`.

Например:

```
makefile
```

```
syms x;
```

```
eqn = x^2 - 4 == 0;
```

```
sol = solve(eqn, x);
```

Как реализовать численный метод дифференцирования в MATLAB? Численный метод дифференцирования в MATLAB можно реализовать с помощью функции `diff()`. Например:

```
dx = diff(x);
```

Как создать собственную функцию в MATLAB и использовать ее?

Собственную функцию в MATLAB можно создать, написав ее код в отдельном файле с расширением `.m`, а затем вызвать из основной программы. Например:

```
function y = myFunction(x)
```

```
y = x^2;
```

```
end
```

8.1. Оценка знаний (академической успеваемости)

осуществляется по 100 балльной системе (шкале) следующим образом:

30 балльная система	100 балльная система	Оценка по буквенной системе	Цифровой эквивалент оценки по GPA	Оценка по традиционной системе
26 - 30	87 – 100	A	4,0	Отлично
24 - 25	80 – 86	B	3,33	Хорошо
22 - 23	74 – 79	C	3,0	
20 - 21	68 - 73	D	2,33	Удовлетворительно
18 - 19	60 – 67	E	2,0	
9 - 17	31 - 60	FX	0	Неудовлетворительно
0 - 8	0 - 30	F	0	

8.2. Технологическая карта дисциплины

Всего	Ауд. часы	СРС, СРСР	1-модуль (90 ч., 30 б.)			2-модуль (90 ч., 30 б.)			Итоговый контроль (ЖТ) (30 б.)			Поощрительный балл (П)	Всего			
			Ауд. ч.		СРС,	Рубежный	Ауд. ч.		СРС,	Рубежный	Лекция			Лаборатория	СРС	Итоговый контроль (ИК)
			Лекц	Прак			Лекц	Прак								
90	4 5	4 5	15	3 0	45	1 5	30	4 5	Рубежный контроль 2 (М2)	3 0	3 0	3 0	30 6	10 6		
Баллы			30	3 0	30	30	3 0	3 0		30 6	3 0	3 0	3 0	30 6	10 6	
Итоги модулей			ТК=(Лек+Лаб+СРС)/3, М1=(ТК1+ТК2+РК1)/3			ТК=(Лек+Лаб+СРС)/3, М2=(ТК3+ТК4+РК2)/3				ИК=(Лек+Лаб+СРС)/3, Экз=М1+М2+ИК+П				100		

3. Тематический план и содержание учебной дисциплины «Практикум решения математических задач на ЭВМ»

	Наименование тем	Кол-во кр лекций
1.	Лабораторная работа №1,2 Интерфейс среды matlab и основы программирования(4 ч)	4

2.	Лабораторная работа 2,3, 4 Простейшие вычисления в matlab	6
3.	Лабораторная работа 5,6,7 Простейшие вычисления в matlab (продолжение)	6
4.	Лабораторная работа 8,9,10 Табулирование функции	6
5.	Лабораторная работа 11,12 Условные операторы	4
6.	Лабораторная работа 13,14 Программирование задач циклической структуры	4
7.	Лабораторная работа 15, 16, 17 Решение задач алгебры и анализа	6
8.	Лабораторная работа 18,19,20 Графика в matlab	6
9.	Лабораторная работа 21,22,23 Работа с массивами в matlab	6
10.	Лабораторная работа 24,25,26 Работа с файлами в matlab	6
11.	Лабораторная работа 27,28,29,30 Решение обыкновенных дифференциальных уравнений в matlab	6
		60

СРСР

№	Тема	Описание	Иновационные компоненты
1	Виртуальная лаборатория численных методов	Студенты разрабатывают и фиксируют эксперименты по решению ОДУ и СЛАУ в онлайн-симуляторе MATLAB Online, сравнивают результаты и визуализируют их в веб-интерфейсе.	AR/VR-модули , облачные симуляции, интерактивная визуализация данных
2	Геймификация решения задач оптимизации	Реализация мини-соревнования: команды соревнуются в скорости нахождения минимума функций, получают бейджи и баллы в LMS за каждую правильно оформленную стратегию.	Геймификация , система бейджей, рейтинги, балансировка нагрузки через цифровую доску достижений
3	Flipped classroom: знакомство с библиотеками Python	До практики студенты смотрят серию коротких видеороликов о NumPy/SciPy, выполняют	Flipped-classroom , видеоматериалы,

```
disp('Zero');
end
```

Как реализовать функцию обратного вызова в MATLAB GUI?

Функцию обратного вызова в MATLAB GUI можно реализовать, добавив соответствующий обработчик событий к графическому элементу. Например:

```
lua
function myCallback(hObject, eventdata, handles)
    % ваш код обработки события
end
```

Как создать панель инструментов в MATLAB GUI? Панель инструментов в MATLAB GUI можно создать с помощью функции `uitoolbar()`. Например:

```
toolbar = uitoolbar(fig);
```

Каким образом можно реализовать взаимодействие с пользователями в MATLAB? Взаимодействие с пользователями в MATLAB можно реализовать с помощью графических элементов интерфейса пользователя и функций обратного вызова. Например:

```
uicontrol('Style', 'pushbutton', 'String', 'Нажми меня', 'Callback', @myCallback);
```

Как использовать структуры данных в MATLAB? Структуры данных в MATLAB можно создавать с помощью функции `struct()`. Например:

```
с
myStruct = struct('field1', value1, 'field2', value2);
```

Как создать и использовать ячейки в MATLAB? Ячейки в MATLAB можно создать с помощью круглых скобок и запятых. Например:

```
makefile
□ myCell = {value1, value2, value3};
```

Как создать и использовать структуры в MATLAB? Структуры в MATLAB можно создать с помощью функции `struct()`. Например:

```
с
myStruct = struct('field1', value1, 'field2', value2);
```

Как создать таблицу данных в MATLAB? Таблицу данных в MATLAB можно создать с помощью функции `table()`. Например:

```
lua
```

css

x = A\b;

Как решить обыкновенное дифференциальное уравнение в MATLAB? Обыкновенное дифференциальное уравнение в MATLAB можно решить с помощью функции ode45(). Например:

less

[t, y] = ode45(@myODE, tspan, y0);

Как реализовать метод наименьших квадратов в MATLAB?

Метод наименьших квадратов в MATLAB можно реализовать с помощью функции polyfit(). Например:

scss

p = polyfit(x, y, n);

Как реализовать численный метод интегрирования в MATLAB? Численный метод интегрирования в MATLAB можно реализовать с помощью функции trapz(). Например:

scss

integral = trapz(x, y);

Как создать интерфейс пользователя в MATLAB? Интерфейс пользователя в MATLAB можно создать с помощью GUI Designer или с использованием функций для создания графических элементов. Например:

appdesigner;

Каким образом можно реализовать цикл в MATLAB? Цикл в MATLAB можно реализовать с помощью ключевых слов for, while или с использованием функций для работы с массивами. Например:

for i = 1:10

disp(i);

end

Какие операторы сравнения существуют в MATLAB?

Операторы сравнения в MATLAB включают в себя >, <, ==, ~=, >=, <=.

Как создать условное выражение в MATLAB? Условное выражение в MATLAB можно создать с помощью ключевых слов if, elseif и else.

Например:

if x > 0

disp('Positive');

elseif x < 0

disp('Negative');

else

№	Тема	Описание	Иновационные компоненты
	за рамками аудиторных занятий	онлайн-тесты в eBilim, а на занятии сразу пишут код под руководством преподавателя.	автоматизированные тесты, микролекции
4	AI-ассистент в поиске и анализе численных методов	Используя ChatGPT-API, студенты генерируют варианты кода для решения задач и затем проводят ревью-сессии, оценивая качество сгенерированных алгоритмов.	ИИ-ассистенты, peer-review, автоматический анализ качества кода
5	Проект «Цифровая школа»: интеграция e-learning решений	Команда проектирует модуль дистанционного обучения для школьников по теме «Интерполяция», внедряет видеолекции, интерактивные тесты и систему личного кабинета.	LMS-модули, мультимедиа, адаптивные тесты, отслеживание прогресса студентов
6	Мобильное приложение-тренажёр по решению систем уравнений	Создают прототип на MIT App Inventor, который генерирует случайные СЛАУ, проверяет решения и выдаёт подсказки, синхронизируя прогресс со «сквозным» аккаунтом LMS.	Мобильные технологии, облачная синхронизация, интерактивные подсказки
7	Хакатон «Большие числа»: параллельное вычисление факториала	За 48 часов студенты оптимизируют расчёт факториала больших n с помощью multiprocessing и GPU-ускорения, презентуют результаты на онлайн-конференции.	Параллельные вычисления, GPU-ускорение, виртуальные конференции, командная работа через Slack/MS Teams
8	Интерактивная визуализация алгоритмов сортировки	Используя D3.js и JupyterLab, студенты создают интерактивный веб-визуализатор работы разных сортировок с возможностью управления скоростью и выбором метода.	Веб-технологии, интерактивные дашборды, JupyterLab, клиент-сайдер визуализация
9	Аналитика активности студентов через learning analytics	Настраивают сбор метрик в LMS (время выполнения заданий, количество попыток), анализируют данные и предлагают улучшения курса на основе дашборда в Power BI.	Learning analytics, дашборды Power BI, автоматический сбор данных, биг-дата подход
10	VR-кейс «Погружение в численные методы»	С помощью очков VR студенты проходят виртуальный «тур» по этапам решения СЛАУ, взаимодействуют с 3D-моделью	VR-среды, 3D-моделирование, интерактивное

№	Тема	Описание	Инновационные компоненты
		матрицы и визуально отлавливают ошибки округления.	обучение, сенсорные контроллеры

СРС

№	Тема самостоятельной работы	Описание	Инновационные компоненты
1	Flipped-lab: изучение функций MATLAB Online	Просмотр видеоруководства по интерфейсу MATLAB Online, выполнение онлайн-теста в LMS, подготовка конспекта часто используемых команд	Flipped-classroom, видеоруководства, автоматизированные тесты в LMS
2	AI-консультант в коде	Использование ChatGPT-API для генерации и оптимизации скриптов решения СЛАУ, последующий анализ и интеграция предложенных улучшений	AI-ассистенты, peer-review
3	Облачный Jupyter-Notebook проект	Разработка и публикация в GitHub–Colab ноутбука для табулирования сложной функции с интерактивными виджетами	Облачные вычисления, интерактивные виджеты, совместный репозиторий
4	Геймификация расчёта погрешностей	Создание в LMS серии заданий-челленджей по оценке численных ошибок с начислением звёзд и бейджей за точность и скорость	Геймификация, бейджи, рейтинги в цифровой платформе
5	Визуализация алгоритма решения ОДУ	Построение интерактивного графика алгоритмов решения ОДУ в D3.js или Plotly, управление параметрами и анимация	Веб-технологии, интерактивные дашборды
6	VR-кейс «Исследование динамики систем»	Подготовка сценария виртуального тура по этапам решения системы дифференциальных уравнений с 3D-моделью матрицы	VR-среды, 3D-моделирование, сенсорные контроллеры
7	Learning analytics: анализ своей успеваемости	Экспорт метрик выполнения заданий из LMS, построение дашборда в Power BI, выявление сильных и слабых тем	Learning analytics, BI-дашборды

```
plot(x, y);
subplot(2, 1, 2);
plot(x, z);
```

Как настроить размер шрифта на графике в MATLAB? Размер шрифта на графике в MATLAB можно настроить с помощью функции `set()` и параметра `'FontSize'`. Например:

```
bash
set(gca, 'FontSize', 12);
```

Как построить трехмерный график в MATLAB? Трехмерный график в MATLAB можно построить с помощью функции `plot3()`, указав значения осей x , y и z . Например:

```
x = [0:0.1:2*pi];
y = sin(x);
z = cos(x);
plot3(x, y, z);
```

Как создать сетку данных в MATLAB? Сетку данных в MATLAB можно создать с помощью функции `meshgrid()`, указав диапазон значений для каждой из осей. Например:

```
less
[X, Y] = meshgrid(-2:0.1:2, -2:0.1:2);
```

Как считать данные из файла в MATLAB? Данные из файла в MATLAB можно считать с помощью функции `load()`, указав имя файла. Например:

```
kotlin
data = load('mydata.txt');
```

Как записать данные в файл в MATLAB? Данные в файл в MATLAB можно записать с помощью функции `save()`, указав имя файла и переменные для сохранения. Например:

```
arduino
save('mydata.txt', 'data');
```

Как найти собственные значения и собственные векторы матрицы в MATLAB? Собственные значения и собственные векторы матрицы в MATLAB можно найти с помощью функции `eig()`. Например:

```
scss
[V, D] = eig(A);
```

Как решить систему линейных уравнений в MATLAB? Систему линейных уравнений в MATLAB можно решить с помощью функции `linsolve()` или обратной операции. Например:

```
y = x^2;
```

```
end
```

Как вызвать функцию в MATLAB? Функцию в MATLAB можно вызвать, указав ее имя и передав нужные аргументы в скобках. Например:

```
результат = myFunction(5);
```

Как создать график в MATLAB? График в MATLAB можно создать с помощью функции plot(), указав значения оси x и y. Например:

```
scss
```

```
x = [0:0.1:2*pi];
```

```
y = sin(x);
```

```
plot(x, y);
```

Как настроить оси графика в MATLAB? Оси графика в MATLAB можно настроить с помощью функций xlabel(), ylabel() и title(), которые позволяют добавить метки к осям и заголовок графика. Например:

```
xlabel('Время');
```

```
ylabel('Амплитуда');
```

```
title('График синуса');
```

Как изменить цвет линии на графике в MATLAB? Цвет линии на графике в MATLAB можно изменить, используя параметр 'color' в функции plot(). Например:

```
arduino
```

```
plot(x, y, 'color', 'r'); % Красная линия
```

Как добавить метку к графику в MATLAB? Метку к графику в MATLAB можно добавить с помощью функции legend(), указав соответствующие метки для каждой линии на графике. Например:

```
less
```

```
legend('sin(x)');
```

Как сохранить график в файл в MATLAB? График в MATLAB можно сохранить в файл с помощью функции saveas(), указав имя файла и формат. Например:

```
scss
```

```
saveas(gcf, 'myplot.png');
```

Как построить несколько графиков на одном рисунке в MATLAB? Несколько графиков на одном рисунке в MATLAB можно построить, используя функции subplot() или hold on. Например:

```
subplot(2, 1, 1);
```

№	Тема самостоятельной работы	Описание	Иновационные компоненты
8	Проект e-learning модуля для школьников	Проектирование мини-курса по решению линейных уравнений в Adobe Captivate или H5P с видеолекциями, интерактивными тестами и симуляциями	LMS-модули, мультимедиа, адаптивные тесты
9	Параллельный вычислительный эксперимент	Разработка скрипта в MATLAB Parallel Toolbox или Python multiprocessing, сравнение времени выполнения и документирование результатов	Параллельные вычисления, GPU-ускорение
10	Мобильный тренажёр на App Inventor	Создание приложения-калькулятора решения квадратных уравнений с пошаговой проверкой и синхронизацией прогресса через Firebase	Мобильные технологии, облачная синхронизация
11	Hackathon-статистика	Участие в мини-хакатоне: публикация MATLAB-проекта на GitLab, настройка CI-сборки и автогенерация отчётов в PDF на GitHub Pages	CI/CD, GitOps, мини-хакатон
12	Инклюзивная визуализация	Разработка цветовых схем и аудио-описаний для графиков MATLAB в соответствии с WCAG, создание отчёта для слабовидящих студентов	Accessibility (WCAG), инклюзивный дизайн
13	Peer-review MATLAB-скриптов	Организация в LMS пула запросов на ревью: обмен скриптами между студентами, предоставление комментариев и оценок в цифровом журнале	Peer-review, цифровой журнал
14	Интерактивный веб-симулятор табулирования	Создание SPA на React.js, подключение к MATLAB-WebSocket для генерации и визуализации табличных данных в реальном времени	React.js, WebSockets, интерактивная визуализация
15	GitOps для MATLAB-проектов	Настройка GitHub Actions для автоматического запуска тестов MATLAB при каждом коммите и публикации PDF-отчёта на GitHub Pages	GitOps, CI/CD, автоматическая генерация документации

10. Учебно-методическое, информационное и материально-техническое обеспечение дисциплины

Основная литература:

1. Гайдук, А.Р. Теория автоматического управления в примерах и задачах с решениями в MATLAB [Электронный ресурс] : учебное пособие / А.Р. Гайдук, В.Е. Беляев, Т.А. Пьявченко. - Санкт-Петербург : Лань, 2017. - 464 с. -

Режим доступа: <https://e.lanbook.com/book/90161>

2. Затонский, А.В. Моделирование объектов управления в MatLab [Электронный ресурс] : учебное пособие / А.В.

Затонский, Л.Г. Тугашова. - Санкт-Петербург : Лань, 2019. - 144 с. -

Режим доступа: <https://e.lanbook.com/book/111915>

3. Квасов, Б.И. Численные методы анализа и линейной алгебры. Использование Matlab и Scilab [Электронный ресурс] : учебное пособие / Б.И. Квасов. - Санкт-Петербург : Лань, 2016. - 328 с. - Режим доступа:

<https://e.lanbook.com/book/71713>

Дополнительная литература:

1. Кудинов, Ю.И. Теория автоматического управления (с использованием MATLAB - SIMULINK) [Электронный ресурс] : учебное пособие / Ю.И. Кудинов, Ф.Ф. Пашенко. - Санкт-Петербург : Лань, 2018. - 312 с. - Режим доступа: <https://e.lanbook.com/book/103140>

2. Гилат А., MATLAB. Теория и практика [Электронный ресурс] / Амос Гилат - Москва : ДМК Пресс, 2016. - 416 с. - ISBN 978-5-97060-183-9 - Режим доступа: <http://www.studentlibrary.ru/book/ISBN9785970601839.html>

3. Перельмутер В.М., Пакеты расширения MATLAB. Control System Toolbox и Robust Control Toolbox [Электронный ресурс] / В. М. Перельмутер - Москва : СОЛОН-ПРЕСС, 2008. - 224 с. - ISBN 978-5-91359-023-7 -

Режим доступа: <http://www.studentlibrary.ru/book/ISBN9785913590237.ht>

11. Вопросы для подготовки к экзамену

Экзаменационные вопросы

Что такое MATLAB и для чего он используется? MATLAB - это высокоуровневый язык программирования и среда разработки, используемая для численных вычислений, визуализации данных, разработки алгоритмов, моделирования и анализа данных.

Как создать переменную в MATLAB? Для создания переменной в MATLAB необходимо указать ее имя, затем присвоить ей значение с помощью оператора равенства. Например:

переменная = значение;

Какие основные типы данных поддерживает MATLAB?

Основные типы данных, поддерживаемые MATLAB, включают в себя числа (целые, вещественные), строки, массивы, структуры и ячейки.

Что такое операторы в MATLAB? Операторы в MATLAB - это символы или ключевые слова, которые выполняют операции над данными. Например, арифметические операторы (+, -, *, /), логические операторы (>, <, ==, ~), операторы присваивания (=), и др.

Как написать комментарий в MATLAB? Комментарии в MATLAB начинаются с символа процента (%) или ключевого слова %%, и продолжаются до конца строки. Например:

shell

% Это однострочный комментарий

%% Это многострочный комментарий

Как создать вектор в MATLAB? Вектор в MATLAB можно создать, перечислив его элементы в квадратных скобках и разделив их запятыми. Например:

вектор = [1, 2, 3, 4, 5];

Как создать матрицу в MATLAB? Матрицу в MATLAB можно создать, перечислив ее строки в квадратных скобках и разделив строки точкой с запятой. Например:

матрица = [1, 2, 3; 4, 5, 6; 7, 8, 9];

Каким образом можно выполнить операции с матрицами в MATLAB? Операции с матрицами в MATLAB могут быть выполнены с помощью арифметических операторов (+, -, *, /) и операторов *элементного умножения* (.*), *деления* ./ и *возведения в степень* (.^).

Что такое индексация массивов в MATLAB? Индексация массивов в MATLAB - это процесс доступа к элементам массива по их позиции в массиве. Индексация начинается с 1, а не с 0, как в некоторых других языках программирования.

Как создать функцию в MATLAB? Функцию в MATLAB можно создать, используя ключевое слово function, указав ее имя, входные и выходные аргументы, а также тело функции. Например:

lua

function y = myFunction(x)