

Внутри каждого скрипта должен быть полный код, при запуске которого в воркспейсе выводятся только требуемые результаты. Промежуточные результаты выводиться не должны. Все скрипты должны быть названы именно так, как требуется. Делать можно как обычные скрипты (.m), так и лайвскрипты (.mlx). Создать скрипт можно нажав правой кнопкой в окошке current folder или на панели сверху под плюсиком new нажать стрелочку.

1. Работа с числами и матрицами. Скрипт NUMBERS.M(.MLX)

Задать любые три числа в переменные a , b , c . Вывести числа d , e , f , g – соответственно сумма всех чисел, произведение всех чисел, наибольшее из чисел и модуль разности a и b .

%% (Обычный скрипт можно делить на блоки двойным знаком комментария и выполнять эти блоки отдельно. Прошу так же, как тут, в ваших скриптах разделять таким символом подзадачи для удобства проверки. Лайвскрипт делится на блоки кнопкой section break или сочетанием клавиш Ctl+Alt+Enter)

Задать матрицы (строки) X и $X0$ натуральных чисел от 10 до 100. "Перевернуть" одну матрицу от 100 к 10 вручную (цикл) и вторую с использованием команды сортировки. Вывести первый и последний элементы каждой матрицы после этого действия. Вывести размеры матриц.

%%

Задать матрицу случайных чисел A размера 3×3 . Задать столбец B из трёх случайных элементов от 5 до 8. Решить уравнения $X1 * A = B^T$ и $A * X2 = B$ через операции $/$ и \backslash . Вывести результаты, в комментариях описать, в чём разница этих двух уравнений. Также решить уравнения $X3 * A = B^T$ и $A * X4 = B$ через обратные матрицы и умножение матриц. Сравнить полученные разными методами результаты.

%%

Задать через команды две матрицы: $Y1$ и $Y2$. Одна из них состоит из единиц, вторая – единичная (обе размерности 5). Записать в переменную $Y3$ произведение матриц (в любом порядке), а в переменную $Y4$ – матрицу, получившуюся поэлементным перемножением исходных матриц. Вывести детерминанты матриц $Y3$ и $Y4$. Затем записать вычисление матриц перемножения и поэлементного перемножения не через простейшие команды, а через циклы, сравнить результаты.

2. Работа с графикой 1. Скрипт GRAPHICS1.M(.MLX)

Построить на одной фигуре графики функций модуль (красным), тангенс (чёрным), корень пятой степени (синим) на отрезке от -10 до 10 с шагом 0.5. Подписать оси, график, включить сетку и легенду. С выводом свойств каждого графика, постараться разобраться в этих свойствах (ваш главный помощник – **Help (F1)**). Сохранить

фигуру в формате MATLAB FIGURE (в окне графика File > Save As, а если пропишите сохранение в коде – будет просто супер).

%%

Построить график кривой в пространстве, где по оси Ох – натуральные числа от 1 до 100, по оси Оу – синус, а по оси Oz – косинус. Подписать оси, график, включить сетку.

%%

Начертить красную пунктирную окружность в полярных координатах через `polar`.

%%

Задать две степенные функции: $y = x^2$ и $y = x^3$ на промежутке от 0 до 8, построить графики. Подписать оси, график, включить сетку и легенду. Толщина линий – 2.5 (толщина линий прописывается в функции `plot`). Не графически найти значение аргумента, при котором разность $x^3 - x^2$ принимает значение больше 100, провести в этом месте вертикальную пунктирную линию толщины 1.5. Вывести ещё два таких же окна с графиками, но в пределах от 0 до пунктирной линии и от пунктирной линии до 10. Всего должно быть три окна с графиками, сохранить все три в формате .PDF и .PNG.

3. Вычисление площадей фигур под графиком. Скрипт S-PLOTS.M(.MLX)

Задать функцию

$$y = 5 \cdot \exp\left[-\frac{(x+3)^2}{4}\right]$$

от -5 до 5 с шагом 0.1. Посчитать площадь фигуры под этим графиком методом трапеций вручную и с помощью функции MATLAB, результаты вывести. Также посчитать площадь под графиком методом Монте-Карло. Суть метода Монте-Карло можно почитать на википедии (статья «Метод Монте-Карло», раздел «Интегрирование методом Монте-Карло»). Сделать симуляцию для 100, 500 и 2000 точек, сравнить результаты с методом трапеций и объяснить (в комментариях). Вывести в графическое окно рассматриваемую область, график функции красной линией толщины 2 и для случая 100 точек все эти точки должны быть также выведены маркером "точка" толщины 6.

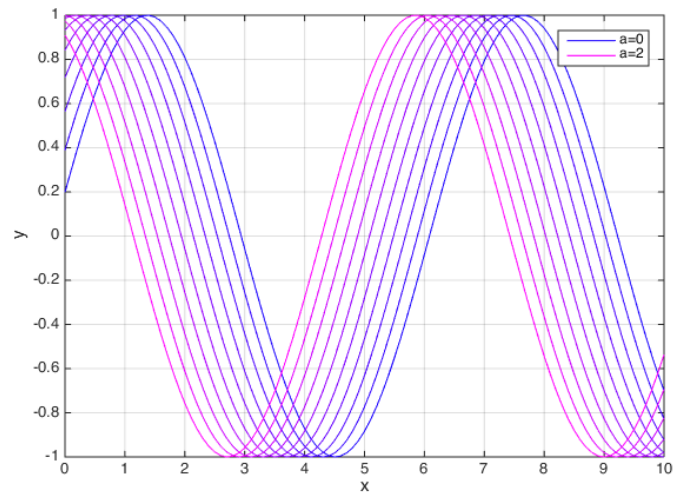
%%

Для предыдущей задачи в том же самом скрипте прописать анимацию выпадения 20 случайных точек на график. *Подсказка: использовать `pause`.*

4. Работа с графикой 2. Скрипт GRAPHICS2.M.

Построить графики функций $y = \sin(x + a)$, где $a \in [0; 2]$. За шаг по a взять 0.25, т.е. всего должно быть нарисовано 9 функций, пусть $a = 0$ соответствует синий цвет

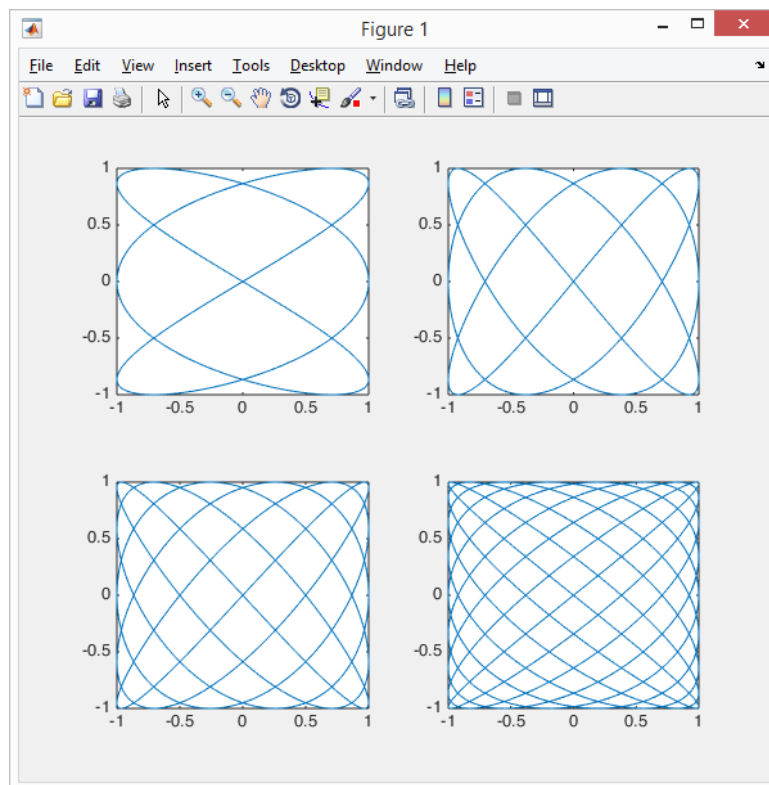
графика, $a = 2$ – розовый. Получить такой график (градиент):



Подписать оси, график, вывести сетку и легенду для конечных значений. *Подсказка: использовать цикл и задание цвета графика через $[r\ g\ b]$.*

%%

Нарисовать несколько фигур Лиссажу (как минимум 4) в одном графическом окне на подграфиках:



Здесь

$$x(t) = \cos(at);$$

$$y(t) = \sin(bt).$$

Подписать каждый из графиков, указать значения для a и b . Также по желанию сделать достаточно плавную анимацию изменения фигур(ы), с параметрами можете поиграться сами, можно, например, ввести фазу в одном сигнале. *Подсказка: subplot*