

2. Работа с графическими объектами в Matlab

Все задания выполнить в одном скрипт-файле.

Структурировать каждое новое задание по разделам-ячейкам (использовать cell mode опцию – в заголовке %%).

Следует использовать глобальные переменные предыдущих заданий, не повторяя команд, если не задана очистка оперативной памяти clear.

Все команды выполнять пошагово, чтобы отследить особенности выполнения

Задание 1

Постройте график функции $y=\cos(x)$, $x\in[0,2\pi]$ и подпишите оси

```
x=0:0.1:2*pi
y=sin(x)
plot(x,y)
xlabel('x')
ylabel('y')
```

Задание 2

Постройте график функции $y=\cos(2x)$, $x\in[0,2\pi]$ красным цветом сплошной линией, выделив опорные точки звёздочкой, полагая, что функция задана строкой.

```
x=0:0.1:2*pi
y=eval('sin(2*x)'),
plot(x,y,'r-*'),
```

1

Задание 3

Постройте графики функций $y=\cos(x)$ и $g=\cos(2x)$, $x\in[0,2\pi]$ красным цветом сплошной линией и пунктирной, поместив оба графика в одни оси и снабдив легендой. Интерактивно поместите текст $\max=1$ у первого максимума

```
%% 3. Постройте графики функций  $y=\cos(x)$  и  $g=\cos(2x)$ ,  $0\leq x\leq 2\pi$ 
красным цветом сплошной линией и пунктирной поместив оба графика
в одни оси и снабдив легендой.
%x=0:0.1:2*pi % можно повторно не выполнять, есть в памяти
y=cos(x)
g=eval('cos(2*x)')
plot(x,y,'r-',x,g,'r--')
legend('y=cos(x)', 'g=cos(2x)')
gtext('max=1') % щелкните в осях в точке расположения надписи
```

Задание 4

Изменить на базе задания 3 толщину пунктирной линии.

```
% 1-й способ
d=get(gca, 'children') % d - двумерный (по числу графиков)
вектор из дескрипторов графиков,
```

```
d=sort(d) % упорядочиваем по убыванию, теперь d(1), d(2)
дескрипторы первого и второго графиков (дескрипторы формируются
так, как в стеке, первый - помещается в конец и имеет меньшую
величину и т.д.)
set(d(2), 'linewidth', 2.5) % изменяем толщину линии, по
умолчанию linewidth=1 ('linewidth'-свойство поля структуры 2.5 -
его значение)
```

% 2-й способ

% создаем новое окно Figure для второго способа:

figure % создается окно со следующим номером или можно

figure(n), n - натуральное

d=plot(x,y,'r-',x,g,'r--');

set(d(2), 'linewidth', 2.5) % здесь синтаксис предыдущей команды
предполагает последовательную нумерацию графиков (первая тройка
- первый график и т.д.)

Задание 5

В новом окне постройте графики функций $y=\cos(x)$, $f=(\cos(x))^2$ и $g=x*\cos(2x)$, на отрезке $2\pi \leq x \leq 2\pi$; предварительно очистив оперативную память. Все графики построить сгенерированным цветом с использованием формата RGB. Вместо легенды оси снабдить заголовком с перечнем функций. Сохранить графический файл с именем myplot.fig (File->Save as (figure))

```
clear
figure
x=-pi:0.1:2*pi
y=cos(x);
f=cos(x).^2;
g=eval(vectorize('x*cos(2*x)')); % Поставьте точку прерывания в
40-ю строку, запустите отладчик;
% в режиме ожидания выделите (при нажатой левой кнопке мыши)
vectorize(g) и щелкните правой
% кнопкой мыши, появится локальное меню, в нем выберите
Evaluate Selection
% - запустится выполнение, проанализируйте результат в командном
окне;
% продолжайте выполнение
YFG=[y' f' g'] % каждый столбец содержит ординаты
соответствующего графика
r=plot(x,YFG); % для x синтаксически не важно это вектор-
строка или столбец
%for k=1:3, set(r(k), 'Color', [k*0.3 1-k*0.3 0.5]), end % если
весь оператор цикла помещаем в одну строку, то каждый
синтаксический элемент в нем отделяем запятой
% в ранних версиях MatLab mycolors{k} - с фигурными скобками
for k=1:3
    set(r(k), 'Color', [k*0.3 1-k*0.3 0.5])
end
title('y=cos(x) f=(cos(x))^2 g=x*cos(2x)')
```

Задание 6.

Построить графики функций $y=\sin(x)$ и $g=x*\exp(x)$, сначала на интервале $(-\pi \leq x \leq \pi)$; затем на $-2*\pi \leq x \leq 2*\pi$.

```
gcf
x=-2*pi:0.2:2*pi;
y=sin(x);
g=eval(vectorize('x*exp(x)'))
subplot(1,2,1), plot(x,y,'b-^',x,g,'m-*') % subplot - строит
матричную структуру графических окон

subplot(1,2,2), plotyy(x,y,x,g) % построение двух графиков с
разным порядком роста, используются две оси ординат
```

Задание 7. Как извлечь файлы графиков из *.fig (в координатной форме)

Выбрать в папке требуемый файл (File->Open (figure)), например, myplot.fig из п.5 (или можно выбрать любые figure, в которых есть оси и графики, чтобы оси стали активными).

Извлечь массивы координат точек имеющихся графиков; gca - активные оси, в которых размещены графики

```
r=get(gca,'children') % gca - графические текущие оси; графики
- потомки осей
xy=[]
for i=1:length(r)
xplots{i}=get(r(i),'xdata')
yplots{i}=get(r(i),'ydata')

xy=[xy(:, :) xplots{i}',yplots{i}']

end

plot(xy(:,1),xy(:,2:2:2*length(r)))
% сохраните матрицу графиков xy в двоичном файле и в текстовом.
```

Задание 8. Используем demos

Полный комплект демо-версий для изучения различных возможностей MatLab доступен, если набрать в командной строке demos .

В командной строке наберите hndlggraf

И изучите в Handle graphics and Line objects, суммируя знания об опциях линий

Задание 9. Построение поверхностей

Построить поверхность $z=x^2+y^2-1$ если $-1 \leq x \leq 1$; $-1 \leq y \leq 1$; двумя способами и поместить в разные окна (с помощью subplot)

```

clear
figure

x=-1:0.1:1
y=-1:0.1:1
[xx,yy]=meshgrid(x,y) % построение сетки в плоскости хоу
z='xx^2+yy^2 -1'
zz=vectorize(z);
pointsurf=eval(zz);
subplot(1,2,1)
surf(pointsurf); % изучить различие этого конструктора и
mesh

subplot(1,2,2)
mesh(pointsurf); % изучить различие этого конструктора и
surf

% интерактивно поменять свойства поверхностей: основное меню
Figure Edit->File Properties

% примитив построения единичной сферы
[sx,sy,sz]=sphere(10) % без визуализации % 10, 40 - число
аппроксимации окружностей x,y,z
sphere(40)

```

в качестве дополнительного задания: познакомьтесь с помощью справки Help с синтаксисом и возможностями использования примитивов line, text; rectangle

4

Задание 10. Постройте графики в общих осях, заданные строками 'sin(x)/x' и 'sin(x)/x^2', для одного из графиков задать цвет в RGB с преобладанием красного

```

clear
ezplot('sin(x)/x'), hold on, r=ezplot('sin(x)/x^2')
set(r,'color',[0.8 0.2 0.5])

```

Задание 11 Самостоятельно

Выберите любой figure с рисунком, зайдите в главное меню окна figure -> Edit -> Figure Properties (Axes Properties или Graph. Object Properties) и отредактируйте максимально разнообразно все детали объектов, а затем сгенерируйте m-file выполненных вами процедур: figure -> File-> Generate m-file. Сохраните файл с новым именем

Разберитесь в деталях. Сформулируйте вопросы, если есть неясности. Используйте в дальнейшем приемы при создании графиков в своих скриптах