Classes 6-7 (4 hours). Graphics in Matlab.

Graphics objects are displayed in a special window that opens with the command `figure.` At the same time, multiple windows can be opened, each one assigned a number. One of the open windows is always active and all commands given on the command line refer to that window.

`figure`	Create a new window by giving it the smallest free number and activating it
`figure (n)`	activates a window with the number n (if any) or creates new ones, assigns n to it and activates them
`close`	closes the active window
`close (n)`	closes window with number n
`close all`	closes all windows
`clf`	clears the active window
`hold on`	retains the current view of the graphical window, any subsequent commands add elements to the active window (eg add new series)
`hold off`	The new graphic replaces the existing one in the active window

Example 1. Create four graphical windows**.**

```
figure
figure
figure(4)
figure
```

The `subplot (mnp)` function creates multiple drawing areas (e.g., for several graphs) within a single graph window. Parameters denote the creation of *m* x *n* subdivisions in *m* rows and *n* columns. The p parameter specifies where the window should be located (from left to right).

Example 2. Create an image window and divide it into four sub-areas (two rows, two columns) by setting it active in the lower right corner.

```
figure
subplot(221)
subplot(222)
subplot(223)
subplot(224)
```

Graphs 2D

`plot(values_x, values_y,'style options')`
where `value_x` and `value_y` are vectors containing x and y coordinates.

Remark. Both vectors x and y must have the same length.

'Style options' are an optional arguments that define color, line style, and style of data point tags. All three elements can be defined together using the form

`color_styleline_markstyle`

Color options	Line style options	style markers
y yellow	- continuous	· dot
m purple	-- dashed	° circle
c turquoise	: dotted	x iks
r red	-. dot-dash	+ cross
g green		* star
b blue		s square
in white		d diamond
k black		v triangle down
		< triangle to the left
		> triangle to the right
		p 5-spoke star
		h 6-spoke star

`xlabel ('text')` - displays text as a description of the x axis of the active graph,

`ylabel ('text')` - displays text as a description of the y axis of the active graph,

`title ('text')` - displays the text as the title of the active graph,

`text (x, y, 'text')` - displays the text at the location specified by x, y,

`legend ('s1, s2, ...)` - displays the legend, s1 is the description of the first graph, s2 the second,

`grid on / off` - enable / disable the secondary coordinate grid display.

Example 3. Draw a red graph of the function $\cos(x)\ e^{\sin(x)}$ in the interval <0, 3π>. Use the marker + and dashed lines. Sign the graph and axes, add a legend, and turn on the grid.

```
x=0:0.1:3*pi;
y=cos(x).*exp(sin(x));
plot(x,y,'r--*');
xlabel('x');
ylabel('y');
title('wykres funkcji cos(x)*exp(sin(x))');
legend('cos(x).*exp(sin(x))');
```

Remark. You can also create an x vector with x = linspace (0.3*pi, 100);

The command `axis([xmin xmax ymin ymax])` is used to scale graphs.

axis('equal')	sets the same scale for both axes
axis('square')	converts a default rectangular frame to a square
axis('normal')	restores default values on axes
axis('off')	removes the frame surrounding the graph and tick marks

Remark. For this command to work, use it after the `plot` command.

`loglog(x,y)`	draws a graph using logarithmic scales on both axes
`semilogx(x,y)`	draws a graph using a logarithmic scale on the x-axis
`semilogy(x,y)`	draws a graph using a logarithmic scale on the y-axis

Overlapping graphs (we use `hold on/off`)

Calling `hold on` freezes the current graph in the graphics window. The next plot generated by the `plot` command is added to the current chart.

3D Graphics

The `plot3` command works similarly to the `plot` command. The appearance of a line, color, style, tags is the same as for a two-dimensional command.

Example 4.

```
t=0:pi/50:6*pi;
x=sin(t);y=cos(t);z=t;
plot3(x,y,z)
axis square;
grid on;
```

Graphs and surface diagrams

Functions for drawing mesh and surface graphs are `mesh` and `surf(Z)`.

To create a surface plot, first create a coordinate grid (x, y) and find the height (z coordinate) of the surface at each point in the grid. You can create a grid of dots over a selected area using the `meshgrid` function.

Example 5. Plot function $z = \frac{xy\,(x^2-y^2)}{x^2+y^2}$, $-3 \le x \le 3, -3 \le y \le 3$, calculating the z values in the area 50x50 in the indicated domain.

```
x=linspace(-3,3,50);
y=x;
[X,Y]=meshgrid(x,y);
Z=X.*Y.*(X.^2-Y.^2)./(X.^2+Y.^2);
mesh(X,Y,Z)
```

We see that we got a grid graph. To get a surface To get a surface we write `surf(X,Y,Z)`

In addition to `plot`, `plot3`, `mesh` and `surf` commands, Matlab allows you to draw graphs of functions given by formula. To do this we use the commands

fplot('function', [xmin xmax])	graph of the function of one variable in the range from xmin to xmax
ezplot('function', [xmin xmax])	automatically adds the title of the plot and describes the x-axis
ezpolar('function a', [xmin xmax])	ezplot command version for polar coordinates
fplot3(xt,yt,zt,[tmin tmax])	plots xt = x(t), yt = y(t), and zt = z(t) over the interval tmin < t < tmax
fcontour('function', xyinterval)	plots the contour lines of the function z = f(x,y) for constant levels of z over the specified interval
fsurf('function',xyinterval)	creates a surface plot of the function z = f(x,y) over the specified interval
fmesh('function',[xmin, xmax, ymin,ymax])	creates a mesh plot of the symbolic expression f(x,y) over the interval [xmin xmax] for x and [ymin ymax] for y
polarplot(theta,rho)	plots a line in polar coordinates, with theta indicating the angle in radians and rho indicating the radius value for each point
bar (x,y)	draws the bars at the locations specified by x
sphere (n)	draws a surf plot of an n-by-n sphere in the current figure
cylinder	generates x-, y-, and z-coordinates of a unit *cylinder*. You can draw the cylindrical object using surf or mesh, or draw it immediately by not providing output arguments
hist(x)	creates a histogram bar chart of the elements in vector x
rose(theta)	creates an angle histogram, which is a polar plot showing the distribution of values grouped according to their numeric range, showing the distribution of theta in 20 angle bins or less

fill(X,Y,C)	creates filled polygons from the data in X and Y with vertex color specified by C
stairs(X,Y)	plots the elements in Y at the locations specified by X. The inputs X and Y must be vectors or matrices of the same size
pie (X,labels)	specifies text labels for the slices. The number of labels must equal the number of slices. X must be numeric
quiver	plot displays velocity vectors as arrows with components (u,v) at the points (x,y)
ellipsoid	generate ellipsoid
pie3(x)	draws a three-dimensional pie chart using the data in X. Each element in X is represented as a slice in the pie chart

For other features, see help Graphics.

Example 5. Vector field of the function

$$z = x^2 + y^2 - 5\sin(xy), \qquad |x| \le 2, |y| \le 2$$

```
x=linspace(-2,2,100);
y=x;
[X,Y]=meshgrid(x,y);
Z=X.^2-5*sin(X.*Y)+Y.^2;
[dx,dy]=gradient(Z,.2,.2);
quiver(X,Y,dx,dy,2)
```

Pie chart

```
x=[807 3701 731 481 349];
kontynent={'Afryka' 'Azja' 'Europa' 'Ameryka Pln' 'Ameryka Pld'};
pie(x,kontynent)
```

Ellipsoid

```
rx=1;
ry=2;
rz=0.5;
cx=1;
cy=2;
xz=3;
ellipsoid(cx,cy,cz,rx,ry,rz)
```

Exercises

Exercise 1. Create a graph of the function $y = \sin 2x$ for $-\pi \leq x \leq \pi$. Then calibrate it so that $0 \leq x \leq \pi/2$ and $0 \leq y \leq 1$.

Exercise 2. Draw a graph of the function e^{x^2} for $0 \leq x \leq 1$ on a linear scale, on a logarithmic scale on the *y*-axis, on a logarithmic scale on the x-axis and the logarithmic scale on both axes. Draw these graphs as subqueries in one graphical window (use the `subplot` function).

Exercise 3. Draw overlapping graphs of the *sin*(*x*) and *cos*(*x*) functions in the interval $(-\pi, \pi)$. Sign the graph and axes, add a legend, draw a red line with the *sin*(*x*) function, and draw a black line using the rhombs with the rhombs.

Exercise 4. Draw a graph of the function $r^2 = 2\sin 5t,\ 0 \leq t \leq 2\pi$. Use the `polar` command.

Exercise 5. Draw the surface of the function $(x-5)^2 - (y-5)^2$ for $x, y \in [1, 10]$. Use the `meshgrid` and `surf` functions.

Exercise 6. Using the function `plot3` draw a graph of the parametric curve defined by the formulas

$$x(t) = t\cos(2\pi t),\ y(t) = t^2\sin(2\pi t)\,, z(t) = t$$

for $t \in [0, 2\pi]$. Turn on the display of the grid lines.

Exercise 7. Draw a 3D pie chart. For example, data detective – 200, manners – 100, adventure – 50, historical – 10, biography – 20.

Exercise 8. Draw a sphere, an ellipsoid, a histogram, and a circular histogram.

Exercise 9. Use the fill command to draw a triangle with coordinates $A = (0,0)$, $B = (2,4)$, $C = (4,0)$, which will be filled with red color.

Exercise 10. Using the function `fplot` draw a graph of the function $f(x) = x\sin(x)$ for $t \in [-2\pi, 2\pi]$.

Exercise 11. Using the function `fplot3` draw a graph of the parametric curve defined by the formulas

$$x(t) = t\cos(2\pi t),\ y(t) = t^2\sin(2\pi t)\,, z(t) = t$$

for $t \in [0, 2\pi]$.

Exercise 12. Open a new graphical window and divide it into two graphs. On the top graph draw the surface of the function $z = \frac{x^2}{2} + 5 - \sin(x)\cos(y)$ for $|x| < 2\pi$ and $|y| < 2\pi$ using the `fmesh` function, and on the bottom draw a contour diagram of this function using the `fcontour` function.

Exercise 13. Using the function `polarplot` draw a graph of the cardioid $r(t) = 1 + \cos(t)$ for $t \in [0, 2\pi]$.