

# Plotting in MATLAB

## Practical Exampels

Computer Techniques Engineering Deapertment

Second\_Level

Lectuer of Subject: Zainab Maejeed

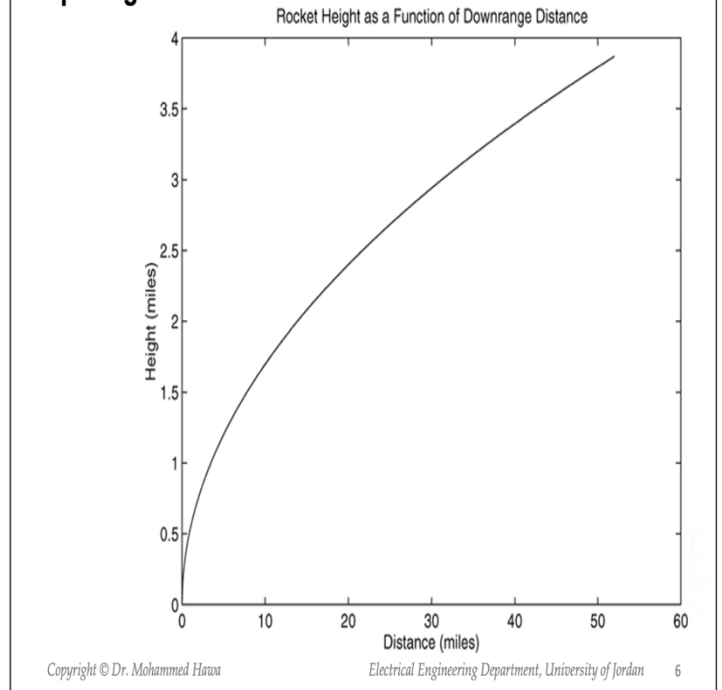
**Example:** Plot  $y = 0.4 \times \sqrt{1.8x}$  for  $0 \leq x \leq 52$ , where  $y$  represents the height of a rocket after launch, in miles, and  $x$  is the horizontal (downrange) distance in miles.

```
>> x = 0:0.1:52;  
>> y = 0.4*sqrt(1.8*x);  
>> plot(x,y);  
>> xlabel('Distance (miles)');  
>> ylabel('Height (miles)');  
>> title('Rocket Height vs. Distance');
```

Notice that for each  $x$  there is  $y$ ; so MATLAB plots one array against another.

Also notice how we added the axes labels and plot title. The resulting plot is shown on the next slide.

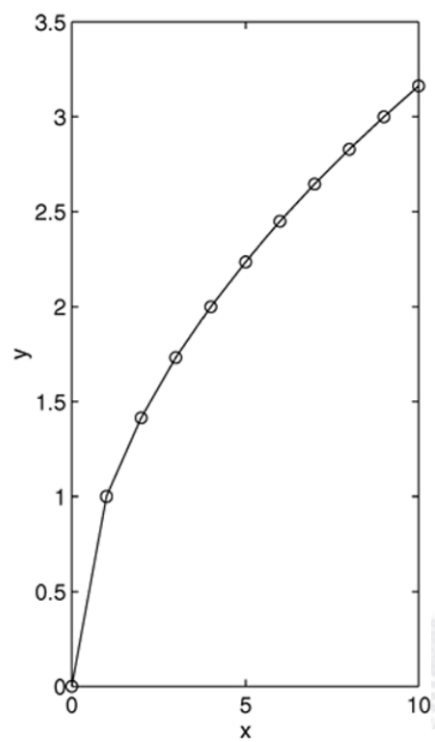
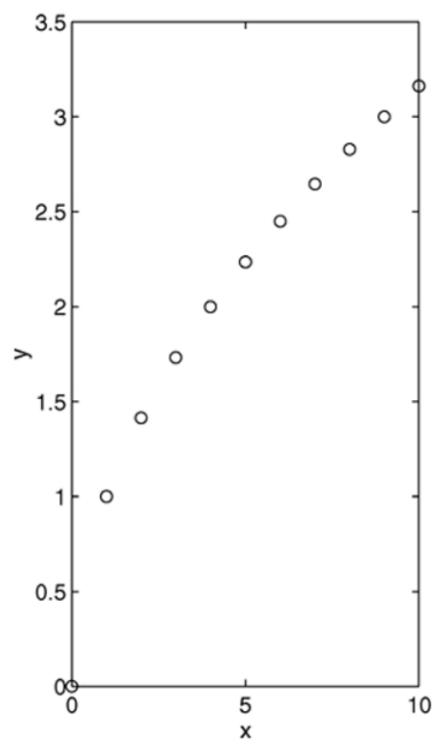
**The autoscaling feature in MATLAB selects tick-mark spacing.**



**The plot will appear in the Figure window. You can use the plot in other applications in several ways:**

- 1. You can print a hard copy of the figure by selecting **File | Print** menu item in the Figure window.**
- 2. You can save the plot to a file to be used later. You can save the plot by selecting **File | Save As** menu item. Possible file formats include: \*.fig (MATLAB format), \*.bmp, \*.eps, \*.jpg, \*.png, \*.tif, \*.pdf, .... Another way to save is **File | Export Setup** that allows specifying options for the output file, then selecting **Export**.**
- 3. You can copy a figure to the clipboard and then paste it into another application using the **Edit | Copy Figure** menu item. For options, use **Edit | Copying Options** menu item.**

## Exercise: How did we use different data markers below?

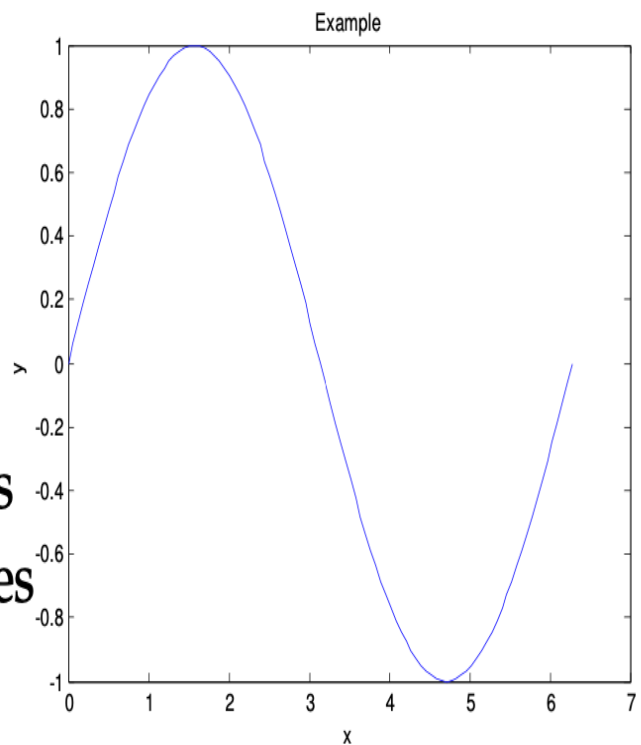


Example2: plot sine wave , where  $0 \leq x \leq 2\pi$ , spacing  $\pi/100$   
Where, x-axis represent X, y-axis represent Y, and title of figure  
(Sine Wave)?

## One Data Set: plot

```
x = 0:2*pi/100:2*pi;  
y1 = sin(x);  
plot(x,y1);  
xlabel('x');  
ylabel('y');  
title('Example');
```

`plot(y1)`: Plots values  
of `y1` versus their indices  
if `y1` is a vector.



Example 2: plot three wave(Sine, Cos, Sine+Cos), by using plot command first the by using hold on , hold off

Where  $0 \leq x \leq 2\pi$ , spacing  $\pi/100$ ,

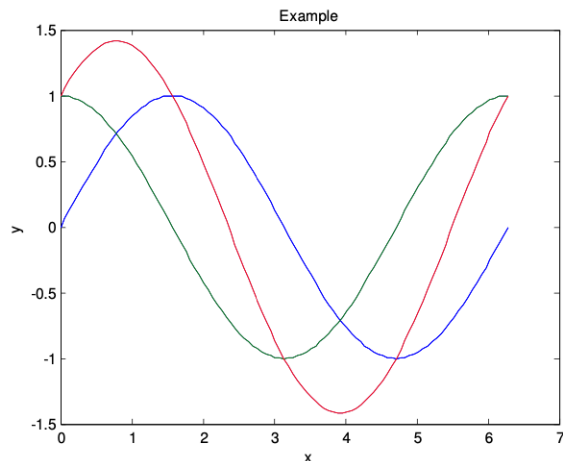
Where, x-axis represent X, y-axis represent Y, Add legend to the figure,

Add title the figure named(and title of figure (Sine &Cos&Sine+cos)?

Solution by using plot command:

## Or better use one plot command

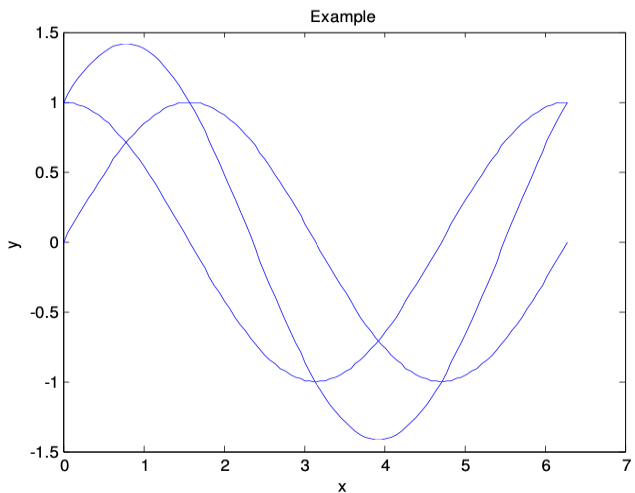
```
x = 0:2*pi/100:2*pi;  
y1 = sin(x);  
y2 = cos(x);  
y3 = sin(x)+cos(x);  
plot(x,y1,x,y2,x,y3);  
xlabel('x');  
ylabel('y');  
title('Example');  
% Notice the auto coloring  
% by MATLAB
```



2. by using hold on , hold, off

## Multiple Data Sets: plot, hold

```
x = 0:2*pi/100:2*pi;  
y1 = sin(x);  
y2 = cos(x);  
y3 = sin(x)+cos(x);  
plot(x,y1);  
hold on;  
plot(x,y2);  
plot(x,y3);  
xlabel('x');  
ylabel('y');  
title('Example');  
hold off;
```

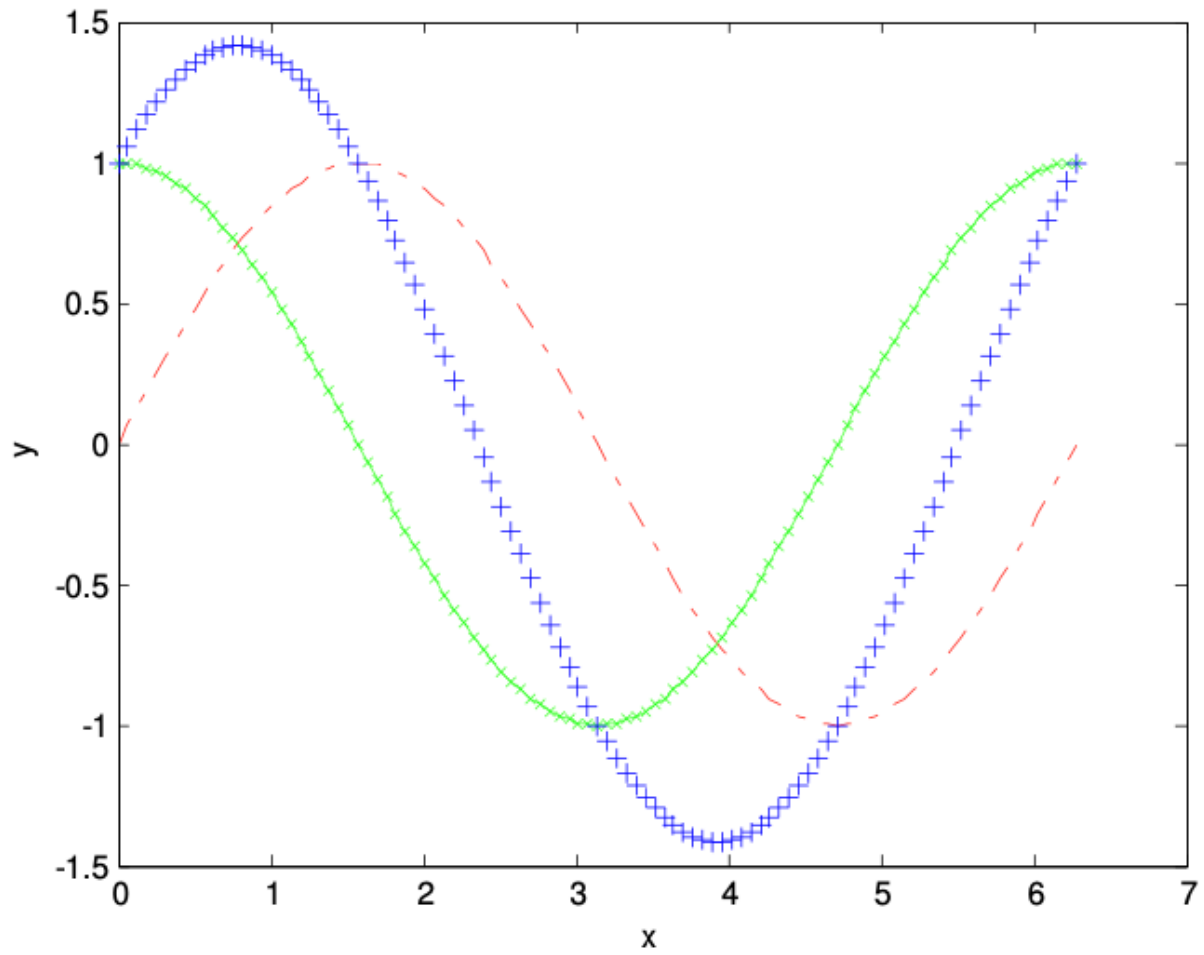


# Colors, Data Markers & Line Types

- You can also specify your own line styles in the plot command.
- For full details enter `help plot` in MATLAB.

|   |         |   |                  |        |         |
|---|---------|---|------------------|--------|---------|
| b | blue    | . | point            | -      | solid   |
| g | green   | o | circle           | :      | dotted  |
| r | red     | x | x-mark           | -.     | dashdot |
| c | cyan    | + | plus             | --     | dashed  |
| m | magenta | * | star             | (none) | no line |
| y | yellow  | s | square           |        |         |
| k | black   | d | diamond          |        |         |
| w | white   | v | triangle (down)  |        |         |
|   |         | ^ | triangle (up)    |        |         |
|   |         | < | triangle (left)  |        |         |
|   |         | > | triangle (right) |        |         |
|   |         | p | pentagram        |        |         |
|   |         | h | hexagram         |        |         |

Homework: plot the following Figure by using plot command?

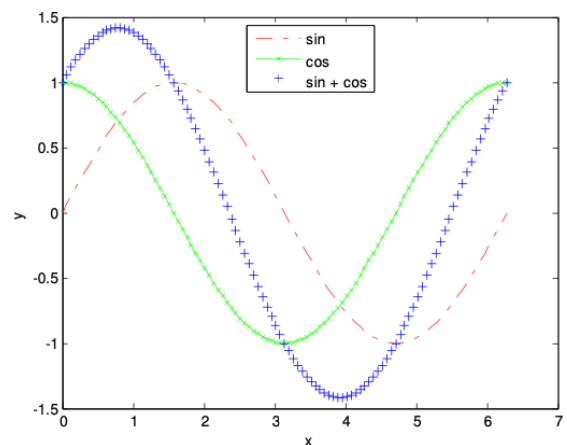
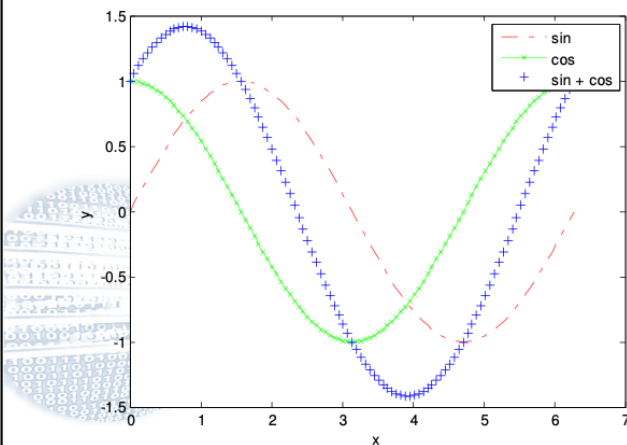


# Legends

- With multiple lines on the same plot it is a good idea to add a legend.

```
legend('sin','cos','sin + cos');  
legend('sin','cos','sin+cos','Location','North');
```

- You can also move the legend with the mouse.



## Labeling Curves and Data

The `legend` command automatically obtains from the plot the line type used for each data set and displays a sample of this line type in the legend box next to the string you selected. The following script file produced the plot in the next slide.

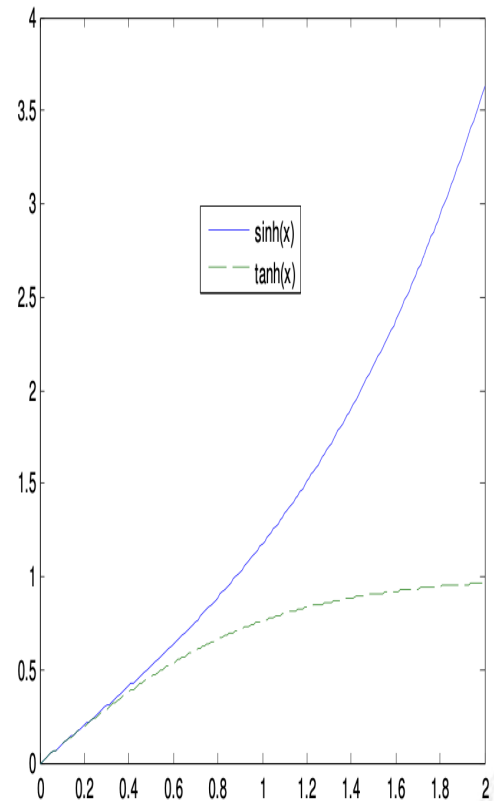
```
x = 0:0.01:2;  
y = sinh(x);  
z = tanh(x);  
plot(x,y,x,z, '--');  
legend('sinh(x)', 'tanh(x)');
```

`gtext('text')`: Places a string in the Figure window at a point specified by the mouse.

`text(x,y,'text')`: Places a string in the Figure window at a point specified by coordinates  $x, y$ .

## Application of the `legend` command.

I moved the legend to an empty space using the mouse.



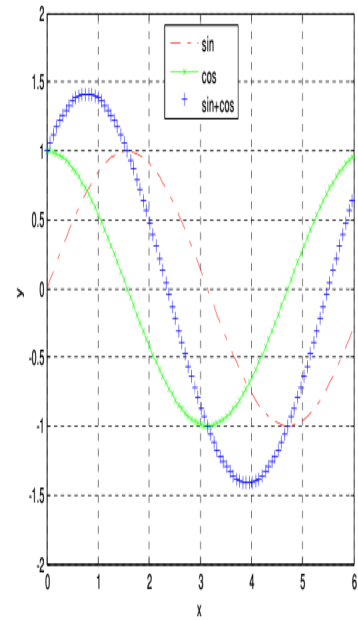
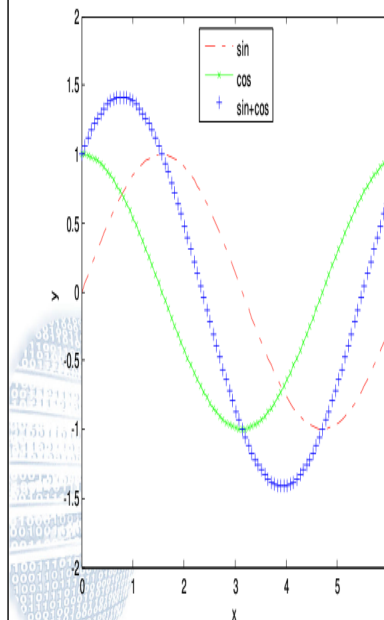
## The `grid` and `axis` Commands

MATLAB will automatically determine the maximum and minimum values for the axes. You can use the `axis` command to override the MATLAB selections for the axis limits. The syntax is `axis([xmin xmax ymin ymax])`. This command sets the scaling for the x- and y-axes to the minimum and maximum values indicated.

The `grid` command displays gridlines at the tick marks corresponding to the tick labels. Type `grid on` to add gridlines; type `grid off` to stop plotting gridlines. When used by itself, `grid` toggles this feature on or off, but you might want to use `grid on` and `grid off` to be sure.

## axis and grid commands

```
axis([0 9 -2 2]);    grid on;  
axis([0 6 -2 2]);    grid off;
```

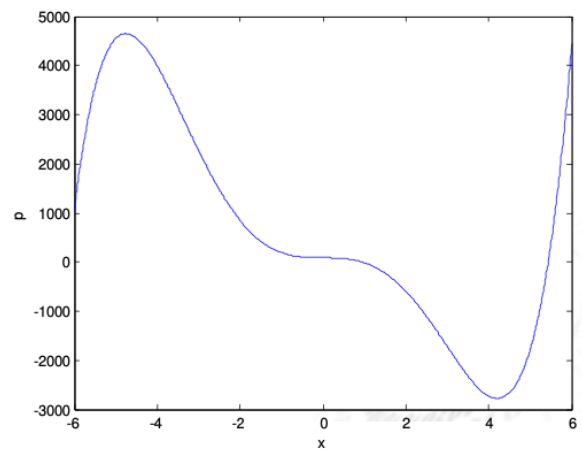


Example: plot the figure by using polyval command?

### Plotting Polynomials with the polyval Function.

To plot the polynomial  $3x^5 + 2x^4 - 100x^3 + 2x^2 - 7x + 90$  over the range  $-6 \leq x \leq 6$  with a spacing of 0.01, you type

```
>> x = -6:0.01:6;  
>> p = [3,2,-100,2,-7,90];  
>> plot(x,polyval(p,x));  
>> xlabel('x');  
>> ylabel('p');
```



Example: plot the following figure

## More Than One Figure Window

- What happens if you enter the following?

```
x = 0:2*pi/100:2*pi;
```

```
y1 = sin(x);
```

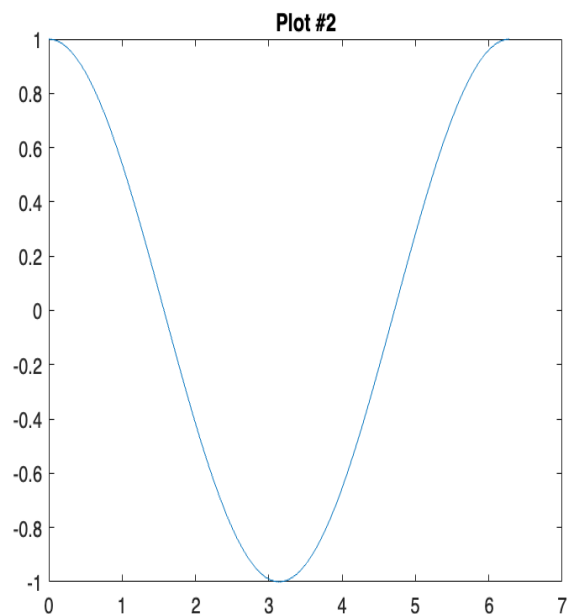
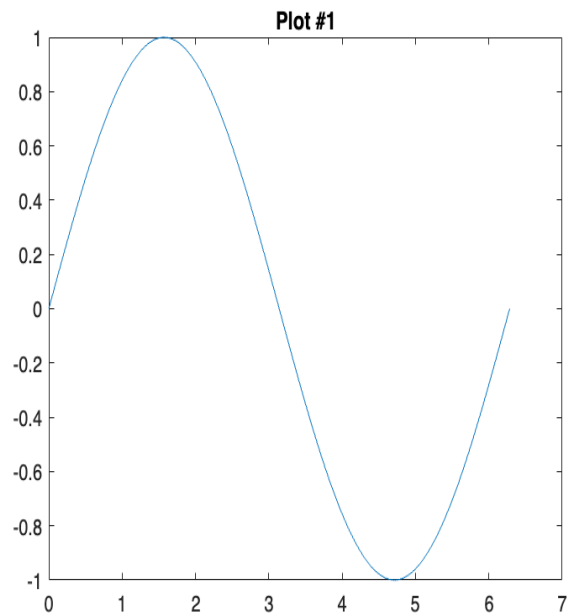
```
y2 = cos(x);
```

```
plot(x,y1);
```

```
title('Plot #1');
```

```
plot(x,y2);
```

```
title('Plot #2');
```



## MULTIPLE FIGURE WINDOWS

- It is possible to open additional Figure Windows and have several of them open (with plots) at the same time. This is done by typing the command `figure`.
- Every time the command `figure` is entered, MATLAB opens a new Figure Window.

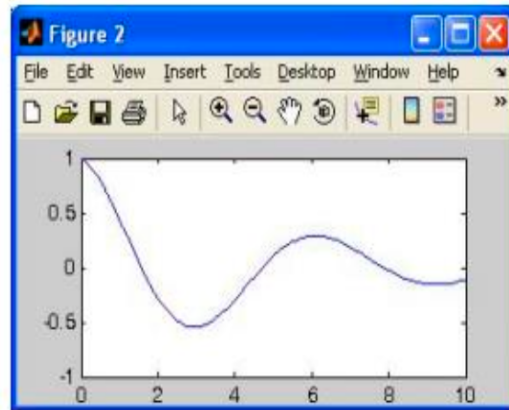
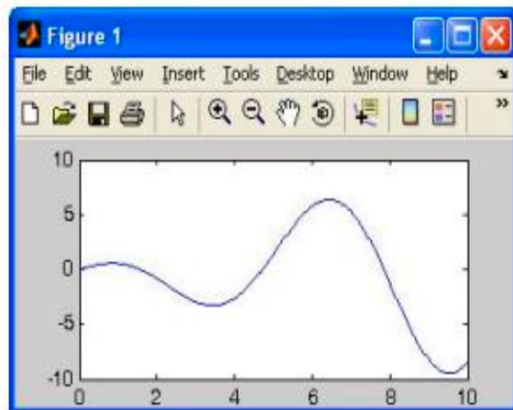
For example, after the following three commands are entered, two Figure Windows that are shown in Figure are displayed.

```
>> fplot('x*cos(x)', [0,10])  
>> figure  
>> fplot('exp(-0.2*x)*cos(x)', [0,10])
```

Plot displayed in Figure 1 Window.

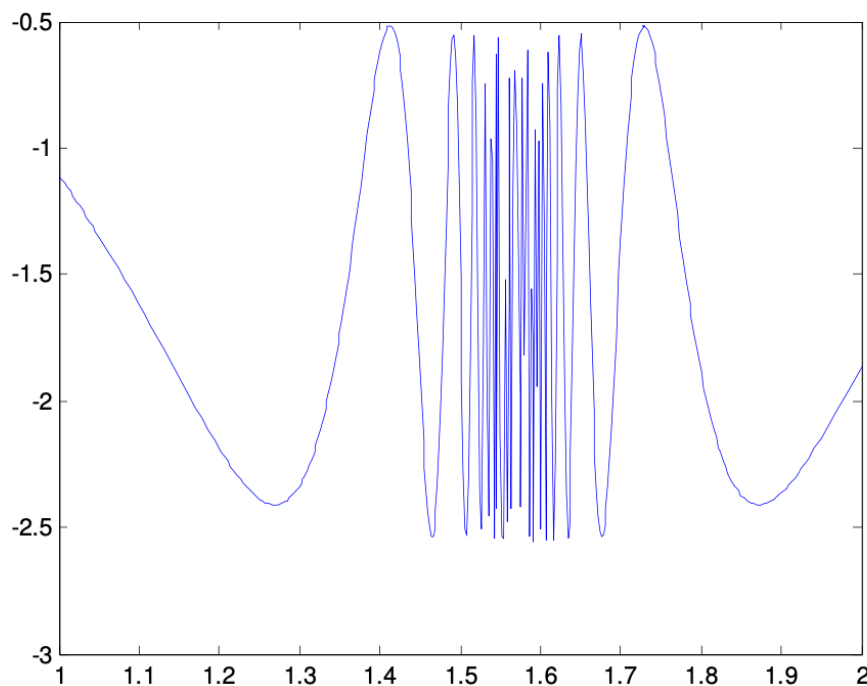
Figure 2 Window opens.

Plot displayed in Figure 2 Window.



The `fplot` command is a “smart” plotting function. Example:

```
f = @(x) (cos(tan(x)) - tan(sin(x)));  
fplot(f,[1 2]);
```



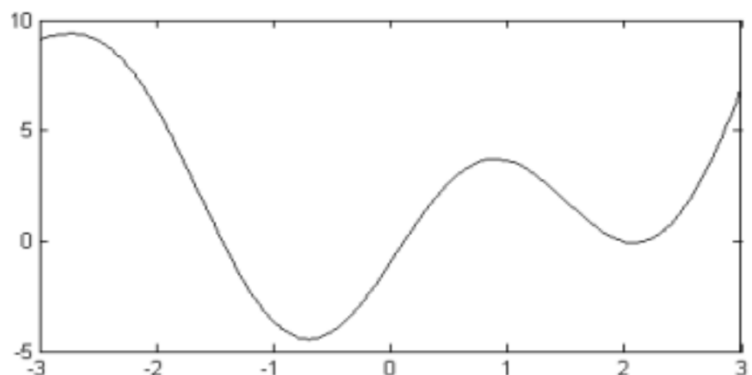
Exapmle : plot the following function by using `fplot` command:

$$y = x^2 + 4\sin(2x) - 1 \text{ for } -3 \leq x \leq 3$$

Can be created with the `fplot` command by typing:

```
>> fplot('x^2+4*sin(2*x)-1',[-3 3])
```

The figure that is obtained in the Figure Window is shown in Figure



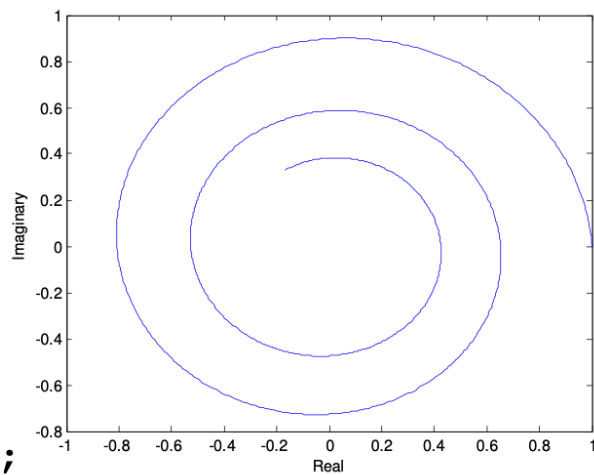
Example: plot the real and imaginary numbers:

## Complex Plot: Real vs. Imaginary

```
n = [0:0.01:10];  
y = (0.1+0.9j).^n;  
plot(y);  
xlabel('Real');  
ylabel('Imaginary');
```

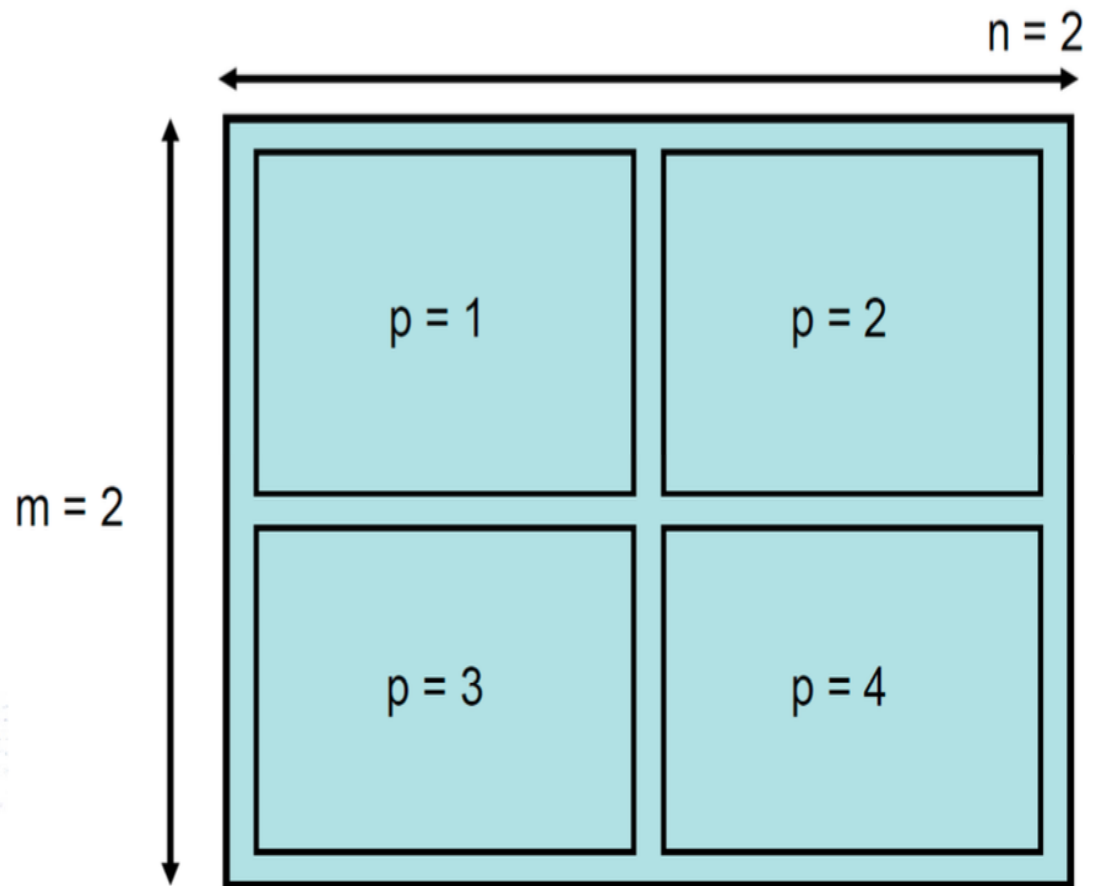
- Similar to:

```
plot(real(y), imag(y));
```



# Subplots

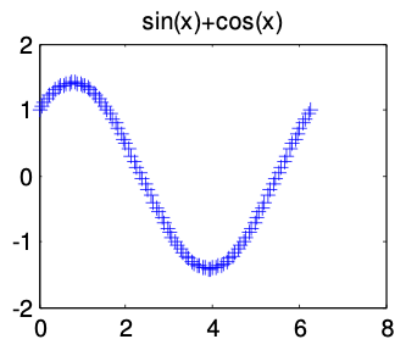
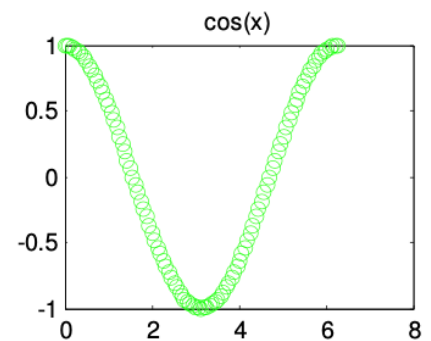
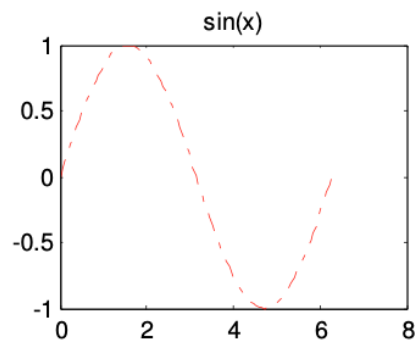
- `subplot(m,n,p)`



Example: by using subplot, plot  $\sin(x)$ ,  $\cos(x)$ ,  $\sin(x)+\cos(x)$ , as show below:

## Example

```
x = 0:2*pi/100:2*pi;  
y1 = sin(x);  
y2 = cos(x);  
y3 = sin(x)+cos(x);  
subplot(2,2,1);  
plot(x,y1,'r-.');  
title('sin(x)');  
  
subplot(2,2,2);  
plot(x,y2,'go');  
title('cos(x)');  
  
subplot(2,2,3);  
plot(x,y3,'b+');  
title('sin(x)+cos(x)');
```



### Homework:

The following script file shows two plots of the functions

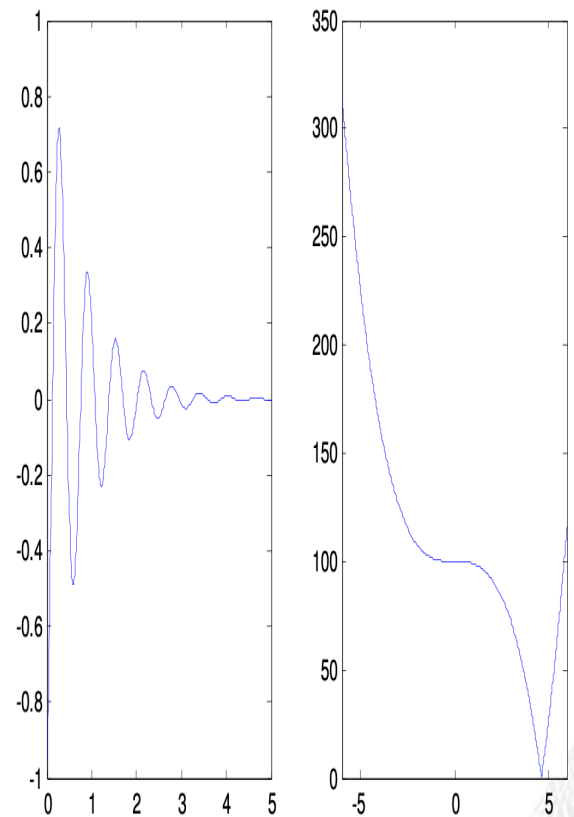
$$y = e^{-1.2x} \sin(10x + 5) \text{ for } 0 \leq x \leq 5$$

$$\text{and } y = |x^3 - 100| \text{ for } -6 \leq x \leq 6.$$

```
x = 0:0.01:5;  
y = exp(-1.2*x).*sin(10*x+5);  
subplot(1,2,1);  
plot(x,y);  
axis([0 5 -1 1]);  
x = -6:0.01:6;  
y = abs(x.^3-100);  
subplot(1,2,2);  
plot(x,y);  
axis([-6 6 0 350])
```

The figure is shown  
on the next slide.

### Application of the subplot command.

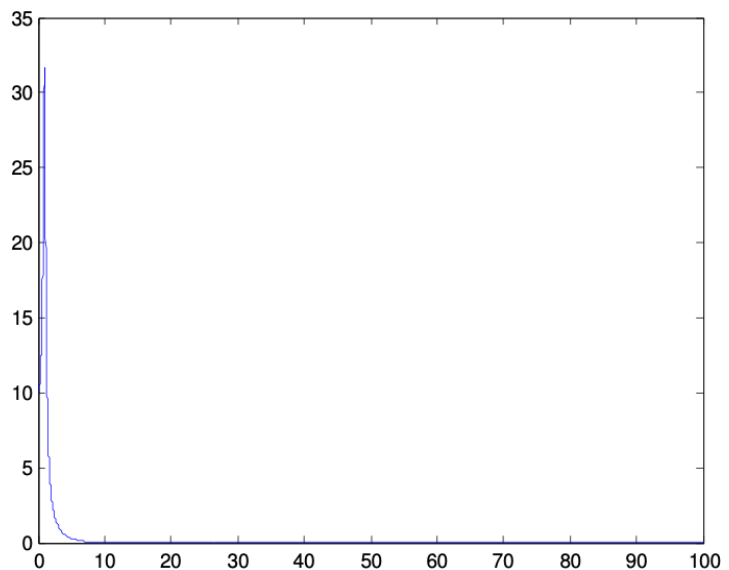


# Log-scale Plots

- Why use log scales? Linear scales cannot properly display wide variations in data values.
- MATLAB has three commands. The appropriate command depends on which axis you want to be a log scale.
- `loglog(x, y)`: both scales logarithmic.
- `semilogx(x, y)`: x-axis is logarithmic and y-axis is rectilinear.
- `semilogy(x, y)`: y-axis is logarithmic and x-axis is rectilinear.
- The syntax is similar to the `plot` command.

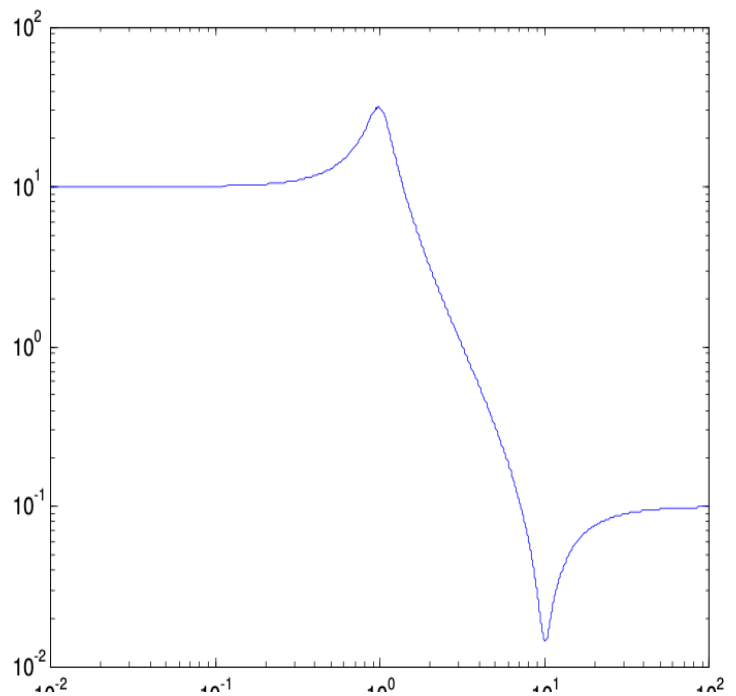
$$y = \sqrt{\frac{100(1 - 0.01x^2)^2 + 0.02x^2}{(1 - x^2)^2 + 0.1x^2}} \quad 0.1 \leq x \leq 100$$

```
x = [0.1:0.01:100];  
y = sqrt((100*(1-0.01*x.^2).^2 ...  
+0.02*x.^2) ...  
./ ((1-x.^2).^2+0.1*x.^2));  
plot(x,y);
```



While when using Log command you can see scales in values

```
x = [0.1:0.01:100];  
y = sqrt((100*(1-0.01*x.^2).^2 ...  
+0.02*x.^2) ...  
./ ((1-x.^2).^2+0.1*x.^2));  
loglog(x,y);
```

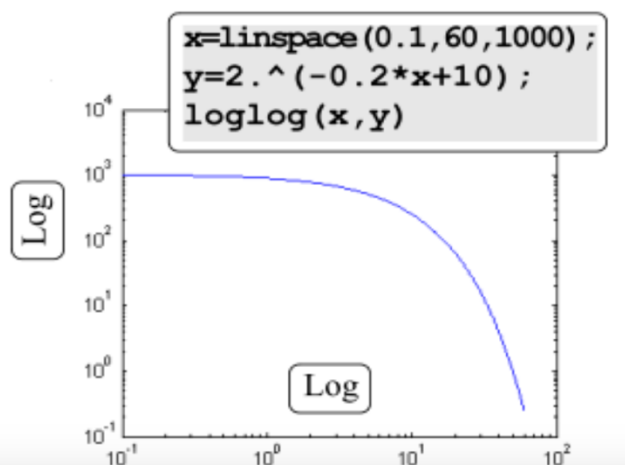
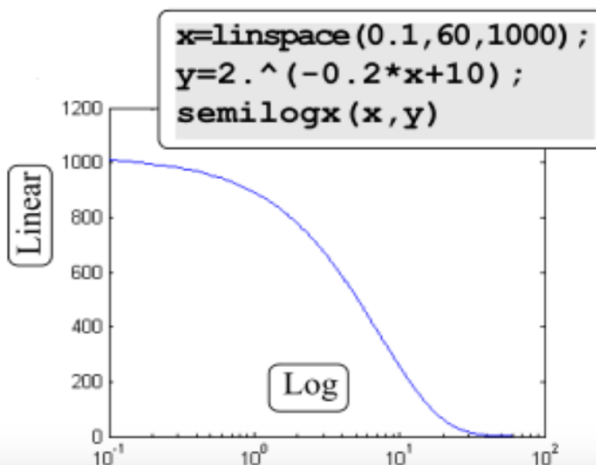
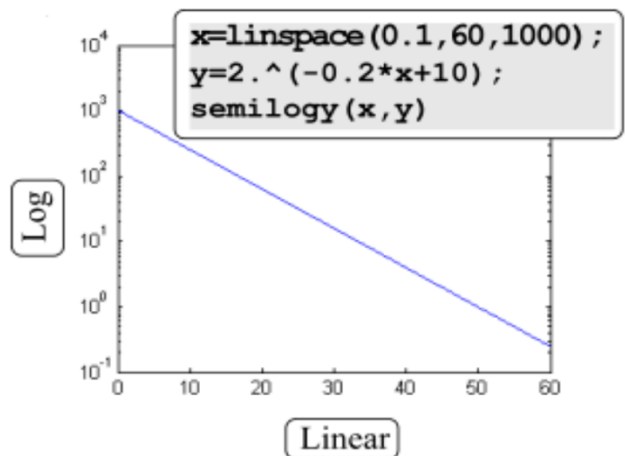
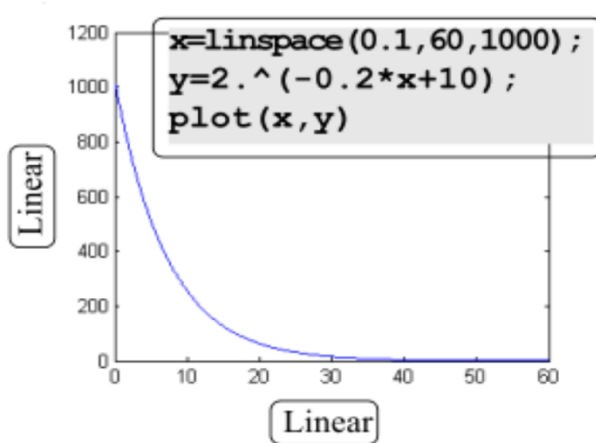


Homework: plot the following function:

$$y = 2^{(-0.2x+10)} \text{ for } 0.1 \leq x \leq 60$$

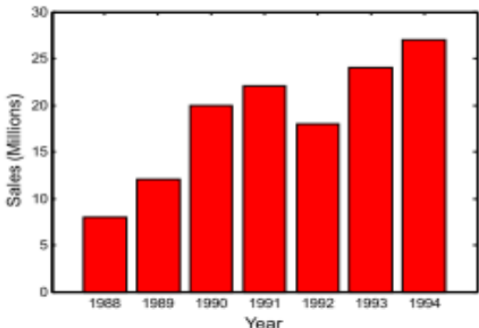
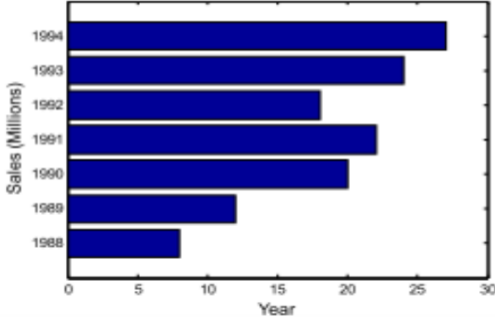
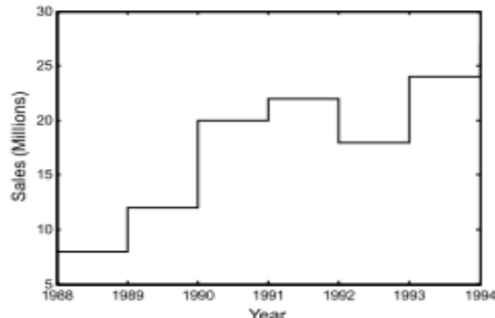
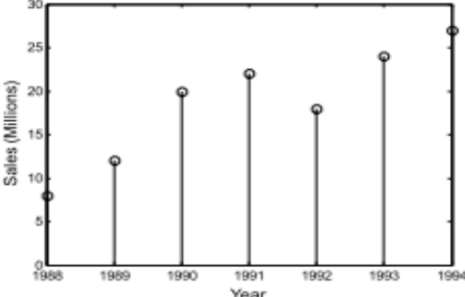
by using

1. Linear plot
2. Semlogx
3. Semiology
4. Loglog
5. Using subplot to show four Figure

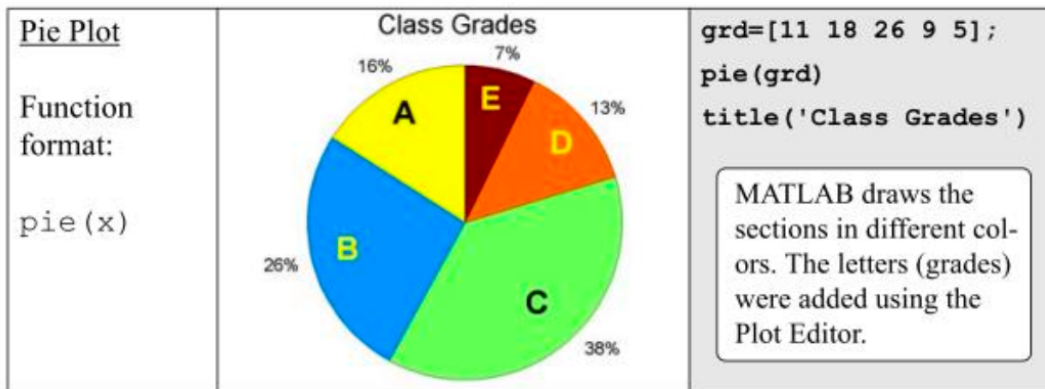


## PLOTS WITH SPECIAL GRAPHICS

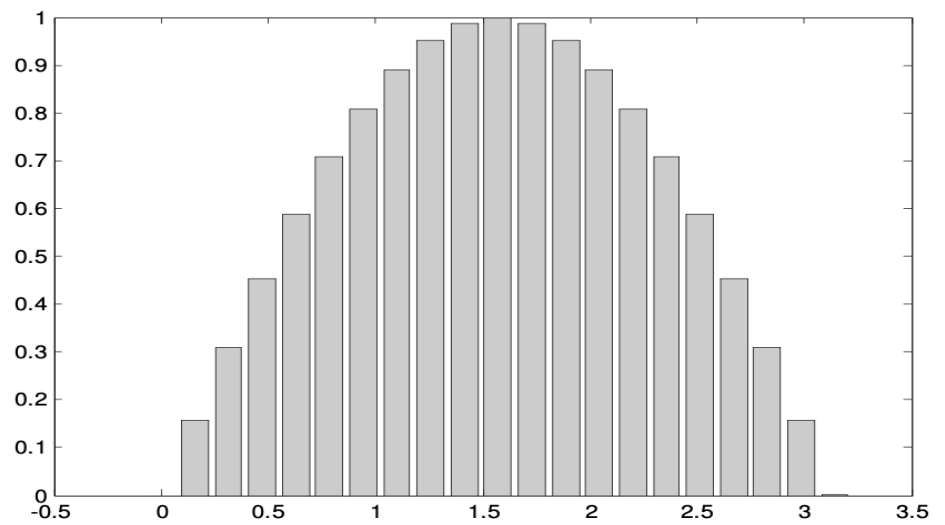
- MATLAB has many options for creating a wide variety of plots. These include bar, stairs, stem, pie plots and many more.
- A complete list of the plotting functions that MATLAB has and information on how to use them can be found in the Help Window.
- Bar (vertical and horizontal), stairs, and stem plots are presented as follows:

|                                                                                  |                                                                                     |                                                                                                                                                                                                                                                                                           |
|----------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <u>Vertical Bar Plot</u><br><br>Function format:<br><br><code>bar(x,y)</code>    |    | <pre> yr=[1988:1994]; sle=[8 12 20 22 18 24 27]; bar(yr,sle,'r') xlabel('Year') ylabel('Sales (Millions)')                     </pre> <div style="border: 1px solid black; padding: 2px; display: inline-block;">                         The bars are in red.                     </div> |
| <u>Horizontal Bar Plot</u><br><br>Function format:<br><br><code>barh(x,y)</code> |   | <pre> yr=[1988:1994]; sle=[8 12 20 22 18 24 27]; barh(yr,sle) xlabel('Sales (Millions)') ylabel('Year')                     </pre>                                                                                                                                                        |
| <u>Stairs Plot</u><br><br>Function format:<br><br><code>stairs(x,y)</code>       |  | <pre> yr=[1988:1994]; sle=[8 12 20 22 18 24 27]; stairs(yr,sle)                     </pre>                                                                                                                                                                                                |
| <u>Stem Plot</u><br><br>Function Format<br><br><code>stem(x,y)</code>            |  | <pre> yr=[1988:1994]; sle=[8 12 20 22 18 24 27]; stem(yr,sle)                     </pre>                                                                                                                                                                                                  |

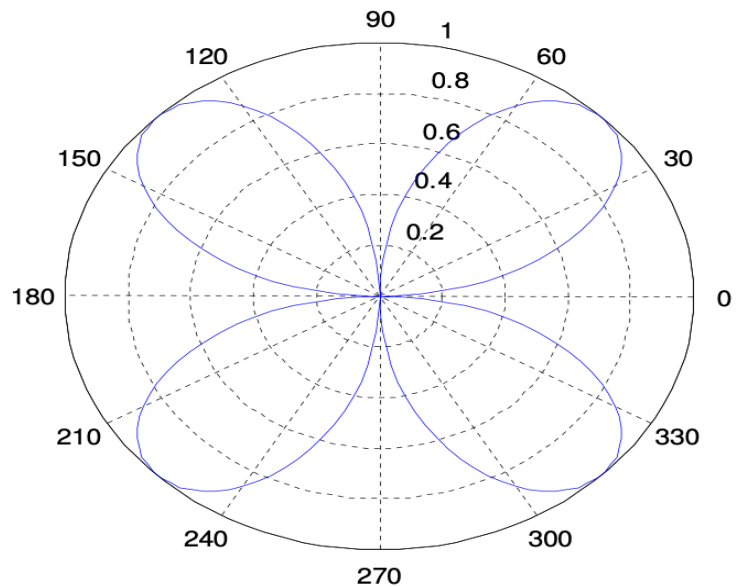
|                    |    |    |    |   |   |
|--------------------|----|----|----|---|---|
| Grade              | A  | B  | C  | D | E |
| Number of Students | 11 | 18 | 26 | 9 | 5 |



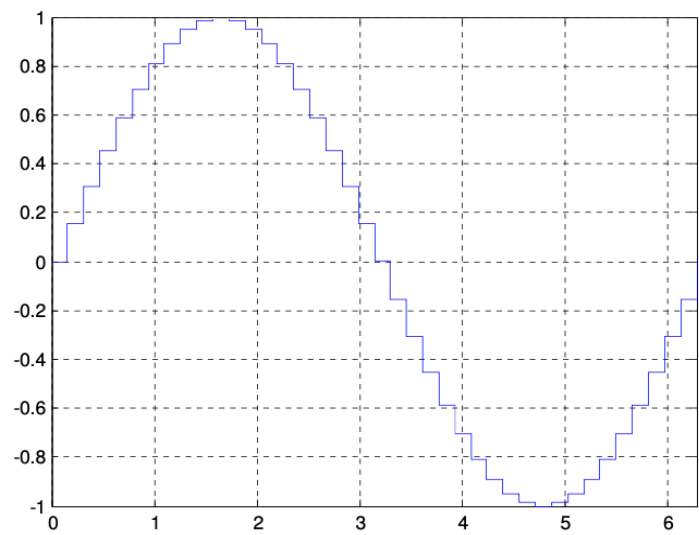
```
x = [0:pi/20:pi];  
bar(x, sin(x));
```



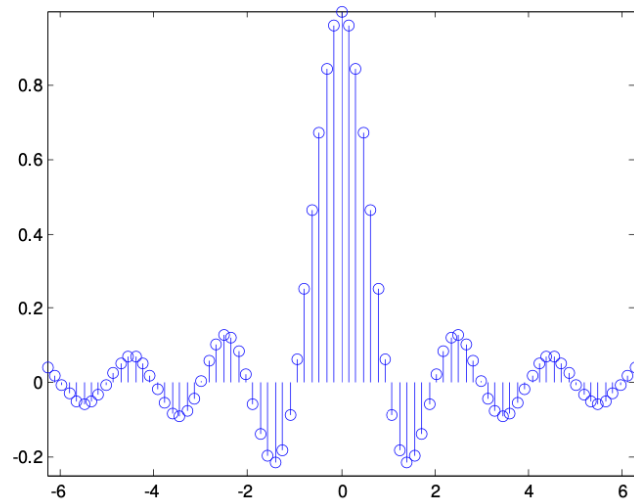
```
theta = [0:pi/90:2*pi];  
polar(theta , sin(2*theta));  
grid;
```



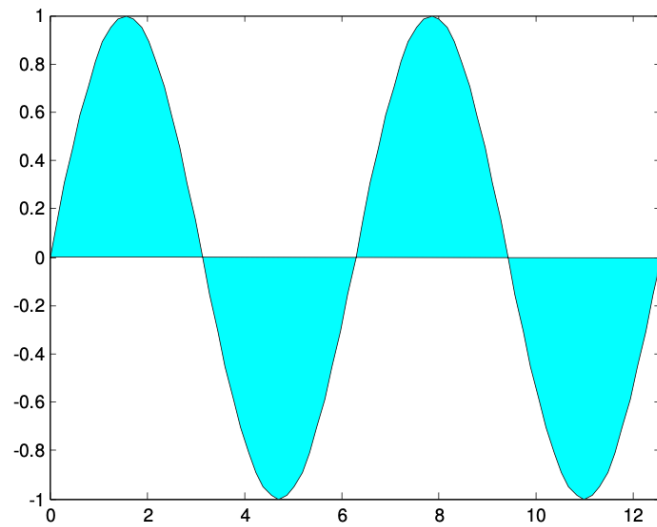
```
x = [0:pi/20:2*pi];  
stairs(x,sin(x));  
grid;  
axis([0 2*pi -1 1]);
```



```
x = [-2*pi:pi/20:2*pi];  
x = x + (~x)*eps;  
y = sin(pi*x)./(pi*x);  
stem(x,y);  
axis([-2*pi 2*pi -.25 1]);
```



```
x = [-2*pi:pi/20:4*pi];  
fill(x,sin(x),'c');  
axis([0 4*pi -1 1]);
```



## Examples

- 1) Generate an overlay plot for plotting three lines

$$y_1 = \sin t$$

$$y_2 = t$$

$$y_3 = t - \frac{t^3}{3!} + \frac{t^5}{5!} - \frac{t^7}{7!};$$

Where  $t = 0 \leq t \leq 2\pi$ , step 0.06. Using

- a. the plot command
- b. the hold command
- c. the line command

### ***Solution:***

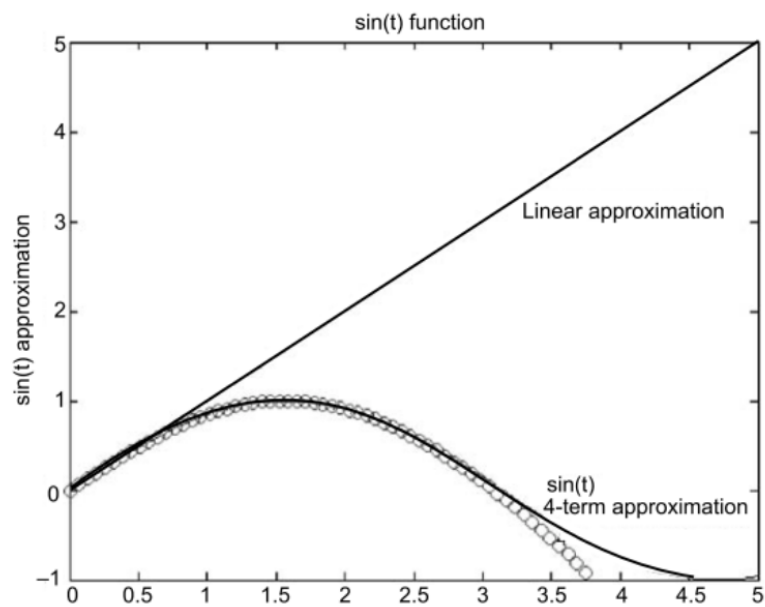
```
% a. using plot command
t=0:0.06:2*pi;
y1=sin(t);
y2=t;
y3=t-(t.^3)/6+(t.^5)/120-(t.^7)/5040;
plot(t,y1,t,y2,'-',t,y3,'o')
axis([0 5 -1 5])
```

```

xlabel('t')
ylabel('sin(t) approximation')
title('sin(t) function')
grid
text(3.5,0,'sin(t) ')
gtext('Linear approximation')
gtext('4-term approximation')

```

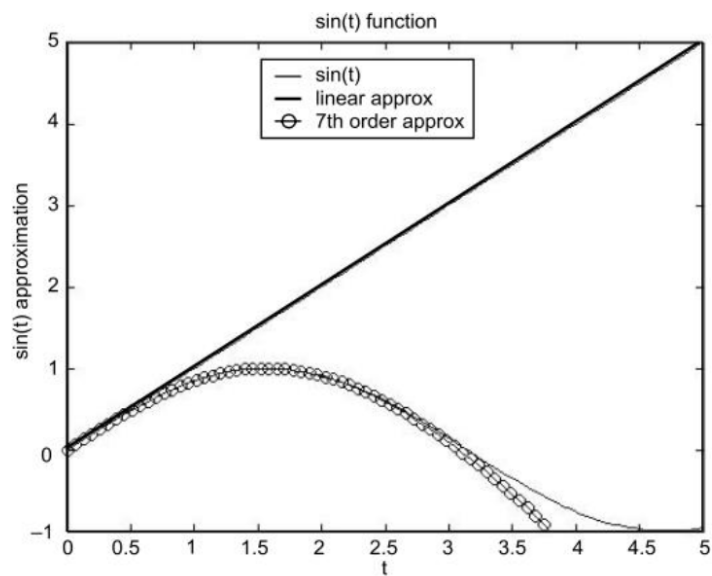
The output is shown in Figure below:



```

% c. using the line command
t=0:0.06:2*pi;
y1=sin(t);
y2=t;
y3=t-(t.^3)/6+(t.^5)/120-(t.^7)/5040;
plot(t,y1)
line(t,y2,'linestyle','-')
line(t,y3,'marker','o')
axis([0 5 -1 5])
xlabel('t')
ylabel('sin(t) approximation')
title('sin(t) function')
legend('sin(t)', 'linear approx', '7th order approx')

```



2) Use the functions for plotting x-y data for plotting the following functions:

(i)  $f(t) = t \cos(t)$   $0 \leq t \leq 10\pi$

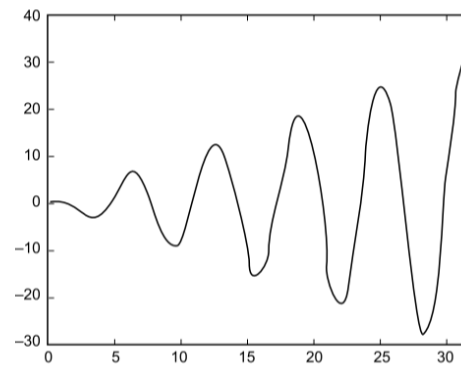
(ii)  $x = e^t$

$y = 100 + e^{3t}$   $0 \leq t \leq 2\pi$  step 0.06

**Solution:**

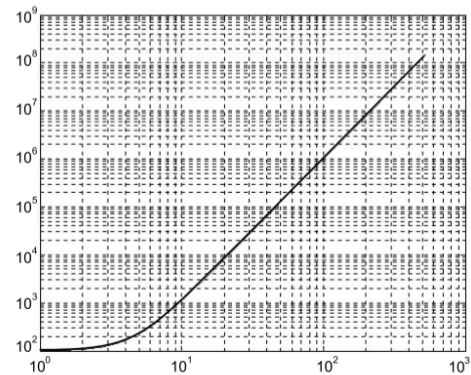
(i)

```
fplot('x*cos(x)', [0 10*pi])
```



(ii)

```
t=0:0.06:2*pi;  
x=exp(t);  
y=100+exp(3*t);  
loglog(x, y)  
grid
```



- 3) Generate a plot of  $y(x) = e^{-0.6x} \cos \omega x$ , where  $\omega = 10$  rad/s, and  $0 \leq x \leq 15$ . Use the colon notation to generate the  $x$  vector in increments of 0.05.

**Solution:**

```
x=[0:0.1:15];  
  
w=10;  
  
y=exp(-0.6*x).*cos(w*x);  
  
plot(x,y)  
  
title('y(x)=e^-0.6x*cos\omegax')  
  
plot(x,y)  
  
title('y(x)=e^-0.6x*cos\omegax')  
  
xlabel('x')  
  
ylabel('y')
```

The output plot is as following:

