Recap
Lecture 1 & 2

- Basic Computations
- Scalars, Vectors, and Matrices
  - Creation
  - Individual element reference/assignment
  - Vectorized element reference/assignment
  - Vectorized manipulations
- Scripts & Functions
- Simple I/O
Basic Plots

**MATLAB**’s `plot` function can be used for simple ”join the dots” x-y plots.

1. EDU\texttt{>> x = [1.5, 2.2, 3.1, 4.6, 5.7, 6.3, 9.4];}
2. EDU\texttt{>> y = [2.3, 3.9, 4.3, 7.2, 4.5, 3.8, 1.1];}
3. EDU\texttt{>> plot(x, y)}
Basic Plots

We could replace \texttt{plot(x,y)} with \texttt{plot(x,y,string)}, where \texttt{string} combines up to three elements that control color, marker and line style.

\begin{verbatim}
1  EDU>> x = [1.5 2.2 3.1 4.6 5.7 6.3 9.4];
2  EDU>> y = [2.3 3.9 4.3 7.2 4.5 3.8 1.1];
3  EDU>> plot(x,y, 'r*--')
\end{verbatim}
Basic Plots

Options for the `plot` command

<table>
<thead>
<tr>
<th>Color</th>
<th>Marker</th>
<th>Line Style</th>
</tr>
</thead>
<tbody>
<tr>
<td>r</td>
<td>Circle</td>
<td>- Solid line (default)</td>
</tr>
<tr>
<td>g</td>
<td>Asterisk</td>
<td>- Dashed line</td>
</tr>
<tr>
<td>b</td>
<td>Point</td>
<td>: Dotted line</td>
</tr>
<tr>
<td>c</td>
<td>Plus</td>
<td>-. Dash-dot line</td>
</tr>
<tr>
<td>m</td>
<td>Cross</td>
<td></td>
</tr>
<tr>
<td>y</td>
<td>Square</td>
<td></td>
</tr>
<tr>
<td>k</td>
<td>Diamond</td>
<td></td>
</tr>
<tr>
<td>w</td>
<td>Upward triangle</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Downward traingle</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Right triangle</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Left triangle</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Five-point star</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Six-point star</td>
<td></td>
</tr>
</tbody>
</table>
Basic Plots

You can add a title and label the axes using the commands `title`, `xlabel`, `ylabel`.

```matlab
EDU>> x = [1.5 2.2 3.1 4.6 5.7 6.3 9.4];
EDU>> y = [2.3 3.9 4.3 7.2 4.5 3.8 1.1];
EDU>> plot(x, y, 'r*--')
EDU>> xlabel('x')
EDU>> ylabel('y')
EDU>> title('Title of the plot')
```

![Graph with title and axis labels]
Basic Plots

Using `hold` to create plots in a single window and using `legend` to label the plots.

```matlab
1 EDU>> x = linspace(0, 2*pi, 40);
2 EDU>> y1 = sin(x);
3 EDU>> y2 = cos(x);
4 EDU>> plot(x, y1, 'ro');
5 EDU>> hold on
6 EDU>> plot(x, y2, 'b+');
7 EDU>> hold off
8 EDU>> legend('sin', 'cos');
9 EDU>> title('sin and cos on one graph');
```
Exercises

Plot the functions $f_1(x) = \sin(\pi \times x)$, $f_2(x) = \sin\left(\frac{3\pi}{2} \times x + \frac{\pi}{6}\right)$ over the interval $[0, 2 \times \pi]$ on the same graph. Use different colors, markers, and line styles for each.
More plots in MATLAB

(a) Bar graph (bar3) (b) Annotated graphs (text) (c) Velocity field (quiver)
M-Files

- Using the Command window is convenient, but can lead to re-typing the same commands over and over.
- An M-file is a text file that has a .m filename extension and contains MATLAB commands. There are two types of M-files:
  - **Script M-files** have no input or output arguments and operate on the variables in the workspace.
  - **Function M-files** contain a function definition line and can accept input arguments and return output arguments, and their internal variables are local to the function.
MATLAB scripts

- A script is a sequence of MATLAB instructions stored in a file.
- The contents of a script can be viewed using type command.
- The script can be run by entering the name of the file at the command line (without the .m extension).
- Warm up task: Create a script file to plot $2 \sin(x)$ and $2 \cos(x - \pi/2)$ in the same figure window and label them using the MATLAB function legend.
The general form of a function definition for a function that calculates and returns a value looks like this: functionname.m

```matlab
1 function outputargument = functionname( ... 
   input arguments)
2  % Comment describing the function
3  Statements here; these must include ... 
   assigning a value to the output argument
```

For example: calcarea.m

```matlab
1 function area = calcarea(rad)
2    % This function calculates the area of ... 
2     a circle
3    area = pi*rad*rad;
```
Exercises

- Given inputs $a$ and $b$, return the sum $a + b$. 
Exercises

- Flip the outermost columns of matrix A, so that the first column becomes the last and the last column becomes the first. All other columns should be left intact. If the input has one column, the output should be identical to the input.

1. Input \( A = \begin{bmatrix} 12 & 4 & 7 \\ 5 & 1 & 4 \end{bmatrix}; \)
2. Output \( B \) is \( \begin{bmatrix} 7 & 4 & 12 \\ 4 & 1 & 5 \end{bmatrix}; \)
Exercises

- Given a positive integer $x$, compute the sum of all integers from 1 to $2^x$. For example, if $x=2$, then $f(x)$ should return $1+2+3+4=10$. 
Exercises

- Write a function which returns every other element of the vector passed in. That is, it returns the all odd-numbered elements, starting with the first.

1. Input \( x = [1 3 2 4 3 5] \)  
2. Output \( y \) is \([1 2 3]\)  
3. Input \( x = [5 9 3 2 2 0 -1] \)  
4. Output \( y \) is \([5 3 2 -1]\)