Administrivia
Homework #11 is due Monday, Nov. 28.
Homework #12 will be due Wednesday, Dec. 7.
Lab #13 is the week after the break, then no lab the last week.
Final examination will be held Dec. 16 at 8:00 a.m. (two hours).
Warmup Quiz
import numpy as np
tmax = 10.0
dt = 0.01
nt = int( tmax/dt ) + 1
x = np.zeros( (nt,) )
for i in range( 0,dt ):
    vx = x[i-1] / np.sin( i )
    x[i+1] = x[i] + vx * dt

Which uncaught error will halt this code?

A ZeroDivisionError
B TypeError
C SyntaxError
D IndexError
import numpy as np
tmax = 10.0
dt = 0.01
nt = int( tmax/dt ) + 1
x = np.zeros( (nt,) )
for i in range( 0,dt ):
    vx = x[ i-1 ] / np.sin( i )
    x[ i+1 ] = x[ i ] + vx * dt

Which uncaught error will halt this code?

A ZeroDivisionError  
B TypeError  
C SyntaxError  
D IndexError
x = np.ones(10)
for i in range(10):
    try:
        ???
    except:
        print('Error on step %d.' % err)
        continue

Which of the following candidates for ??? would not produce an error message?

A  x += x[i+1]
B  x[i] /= 0
C  x[-i-1] = sum(x[:i])
D  x[10-i] = sum(x[:i])
Question #2

```python
x = np.ones(10)
for i in range(10):
    try:
        ???
    except:
        print('Error on step %d.' % err)
        continue
```

Which of the following candidates for `??` would **not** produce any error message?

A. `x += x[i+1]` **index error**
B. `x[i] /= 0` *(surprise!)*
C. `x[-i-1] = sum(x[:i])` *(surprise!)*
D. `x[10-i] = sum(x[:i])` **index error**
Common exceptions

- SyntaxError
- NameError
- TypeError
- ValueError
- IOError
- IndexError
- KeyError
- ZeroDivisionError
- IndentationError
- Exception
Why MATLAB?

- Designed for engineering.
- Excellent documentation: MATLAB Central.
- Ideal applications:
  - Linear algebra
  - Control dynamics
  - Numerical analysis
  - Image processing
- Many toolboxes available.
What is MATLAB?

- Programming language + environment.
- Proprietary, owned and maintained by MathWorks.
- Dates from late 1970s, under active development.
- Was an influence on NumPy/MPL, so will be familiar.
Basics

- Literals, variables, operators
  
  \[ 4 \times 3 \]
  
  Expressions

  \[ a = 3 \times 2 \]
  \[ b = 1 + a \]

  Semicolon suppresses output (mutes): ;

  \[ b = b + 2; \]

  ans is default result.

  \[ a / 4 \]

  disp displays the value only.

  disp( ans );
MATLAB implements:
- integers
- floating-point numbers
- complex numbers

in 8-, 16-, 32-, and 64-bit versions.

`whos` shows type, value of all variables in workspace.
Arrays are the fundamental type in MATLAB:

```
a = [ 1 2 3 ];
b = a( 1 );
```

MATLAB counts from one, not zero!
More dimensional arrays use semicolons to separate rows:

A = [ 1 2 3 ; 4 5 6 ];

Arrays are indexed using parentheses and commas:

a = A( 1,2 );

Helper functions are available:

B = ones( 3,3 ) + eye( 3,3 ) + zeros( 3,3 );
Which of the following will produce this array?

A: \[
\begin{bmatrix}
1 & 1 & 1 \\
2 & 2 & 2
\end{bmatrix}
\]

B: \[
\begin{bmatrix}
1 & 1 & 1 \\
2 & 2 & 2
\end{bmatrix}
\]

C: \[
\begin{bmatrix}
1 & 2 \\
1 & 2 \\
1 & 2
\end{bmatrix}
\]

D: \[
\begin{bmatrix}
1 & 2 \\
1 & 2 \\
1 & 2
\end{bmatrix}
\]

E: \[
\begin{bmatrix}
\begin{bmatrix} 1 & 1 & 1 \end{bmatrix}, \begin{bmatrix} 2 & 2 & 2 \end{bmatrix}
\end{bmatrix}
\]
Which of the following will produce this array?

A \[ \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \end{bmatrix} \]

B \[ \begin{bmatrix} 1 & 1 & 1 \\ 2 & 2 & 2 \end{bmatrix} \]

C \[ \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix} \]

D \[ \begin{bmatrix} 1 & 2 \\ 1 & 2 \\ 1 & 2 \end{bmatrix} \]

E \[ \begin{bmatrix} [1 & 1 & 1] \\ [2 & 2 & 2] \end{bmatrix} \]
\[ A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \]

Which of the following will access 4 in this array?

A) \( A(1,0) \)
B) \( A[2,1] \)
C) \( A(2,1) \)
D) \( A(1)(0) \)
Question

\[ A = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \]

Which of the following will access 4 in this array?

A  \( A(1,0) \)
B  \( A[2,1] \)
C  \( A(2,1) \)
D  \( A(1)(0) \)
% basic mathematics:
A = ( ones( 3,3 ) + 1 ) / 2
B = sin( ones( 3,3 ) * pi )
C = B'  % transpose with '

% matrix multiplication:
D = eye( 3,4 ) * ones( 4,5 ) * pi


\[
\begin{pmatrix}
2 & 1 \\
1 & 2
\end{pmatrix}
\]

Which of the following will produce this array?

A. \(3 \cdot \text{ones}(2, 2) - 2 \cdot \text{eye}(2, 2)\)
B. \(2 \cdot \text{ones}(2, 2) + \text{eye}(2, 2)\)
C. \(3 \cdot \text{ones}(2, 2) - \text{eye}(2, 2)\)
D. \(\text{ones}(2, 2) + \text{eye}(2, 2)\)
Which of the following will produce this array?

A \( 3 \times \text{ones}(2,2) - 2 \times \text{eye}(2,2) \)
B \( 2 \times \text{ones}(2,2) + \text{eye}(2,2) \)
C \( 3 \times \text{ones}(2,2) - \text{eye}(2,2) \)
D \( \text{ones}(2,2) + \text{eye}(2,2) \)
% concatenating arrays
A = [ eye(3,4), eye(3,5);
     ones(2,4), ones(2,5) ]
How can we produce this array?

A \[ \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix} \]

B \[ \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \]

C \[ \begin{bmatrix} 1 & 3 & 5 \\ 2 & 4 & 6 \end{bmatrix} \]

D \[ \begin{bmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{bmatrix} \]
How can we produce this array?

A \[
\begin{bmatrix}
1 & 3 & 5 \\
2 & 4 & 6
\end{bmatrix}
\]

B \[
\begin{bmatrix}
1 & 2 \\
3 & 4 \\
5 & 6
\end{bmatrix}
\]

C \[
\begin{bmatrix}
1 & 3 & 5 \\
2 & 4 & 6
\end{bmatrix}
\]

D \[
\begin{bmatrix}
1 & 2 \\
3 & 4 \\
5 & 6
\end{bmatrix}
\]
MATLAB uses `.m` files for two purposes: scripts and functions.

Comments are indicated as follows:

```plaintext
% this is a comment
{%
   this is a block comment
%
```
Use the built-in editor to create these.
Make sure you have the correct working directory.
Scripts contain regular commands in order of execution.
Functions

- Functions must be located in a file of the same name as the function.

```matlab
function [ output variables ] = function_name( input variables )
% ...
end
```

- No explicit `return` statements—rely on values in output variable list.
\[ T_F = \frac{180}{100} T_C + 32 \]

File TempC2F.m:

```matlab
function \[ Tf \] = TempC2F( Tc )
    Tf = Tc * ( 180/100 ) + 32;
end
```
Strings

- Indicated with single quotes (only!).

```matlab
s = 'XFEM';
```

- Print formatted strings with `sprintf`:

```matlab
sprintf( '%f %f', sin(pi/3), cos(pi/4) );
```
Matrix v. element operations

- “Matrix dimensions must agree.”
- It is sometimes necessary to distinguish elementwise operations and matrix operations.

```matlab
A = 2 * ones( 2,2 )
B = A .* eye( 2,2 )
C = A * eye( 2,2 )
```

- These are distinguished by a dot . in front of the operator.
- We won’t emphasize this but frequently you must distinguish.