**ENR 261 Spring 2023 Chapter 9 & 10 Homework**

**General Instructions:**

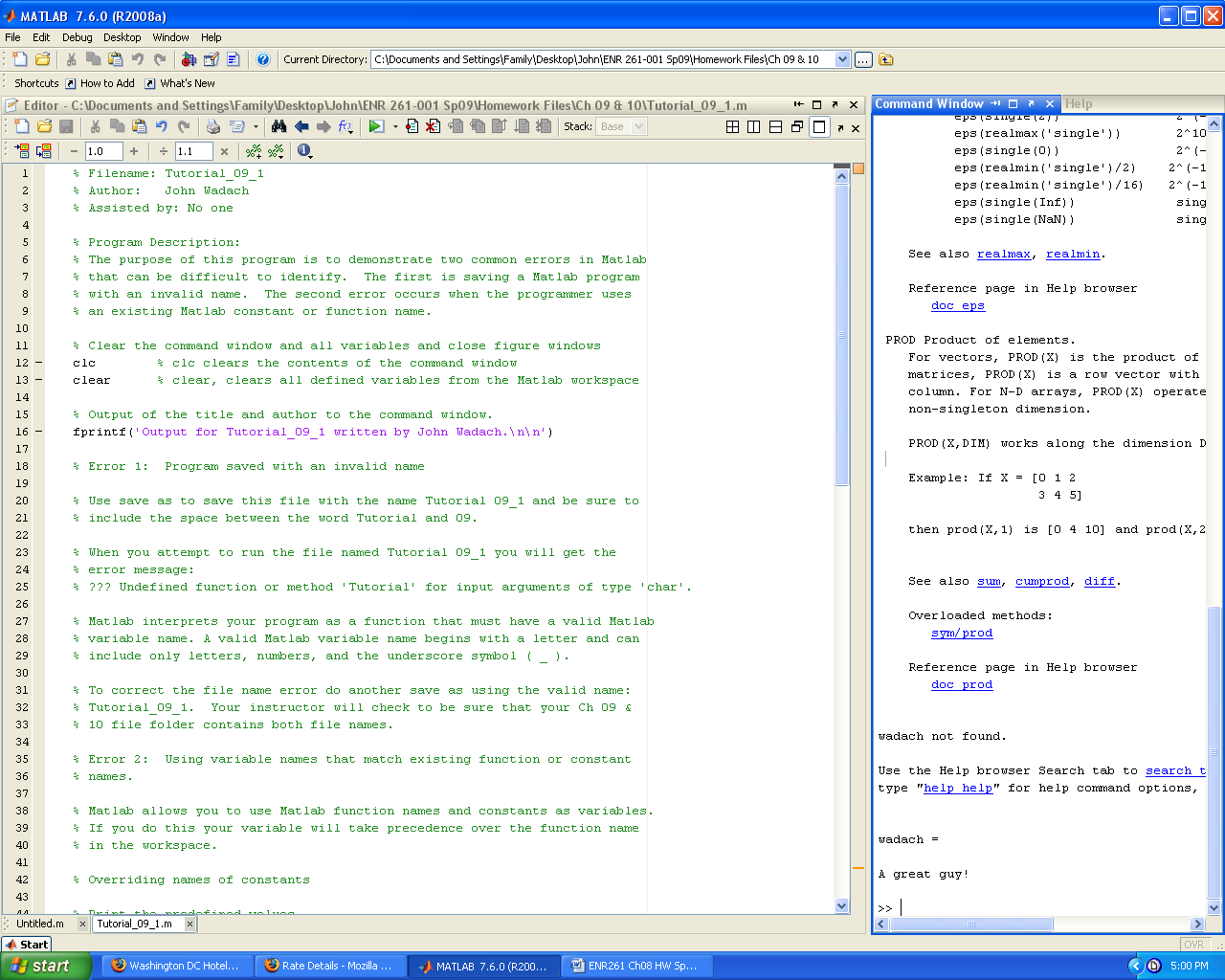
Save your all your Matlab files for this chapter in the folder named **Ch09\_10** located inside your local repository on your USB Memory Stick. When finished be sure to add, commit, and push your changes to your remote repository on GitHub.

**Assigned Exercises**

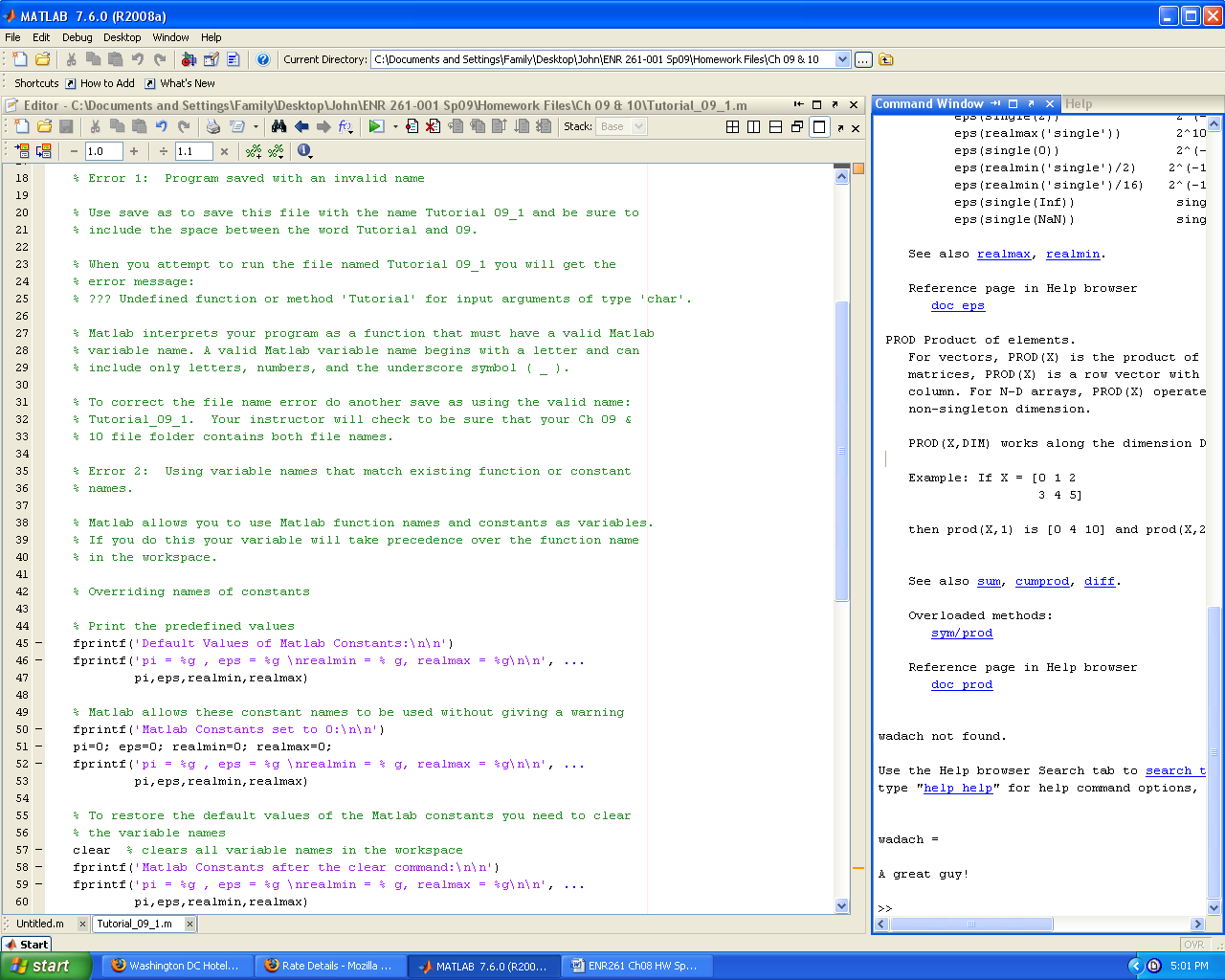
1. Recreate all of the following script files and be sure to save them in your local repository on your USB memory stick, commit the changes and push them to GitHub.

2. Use the required file names for each script file.

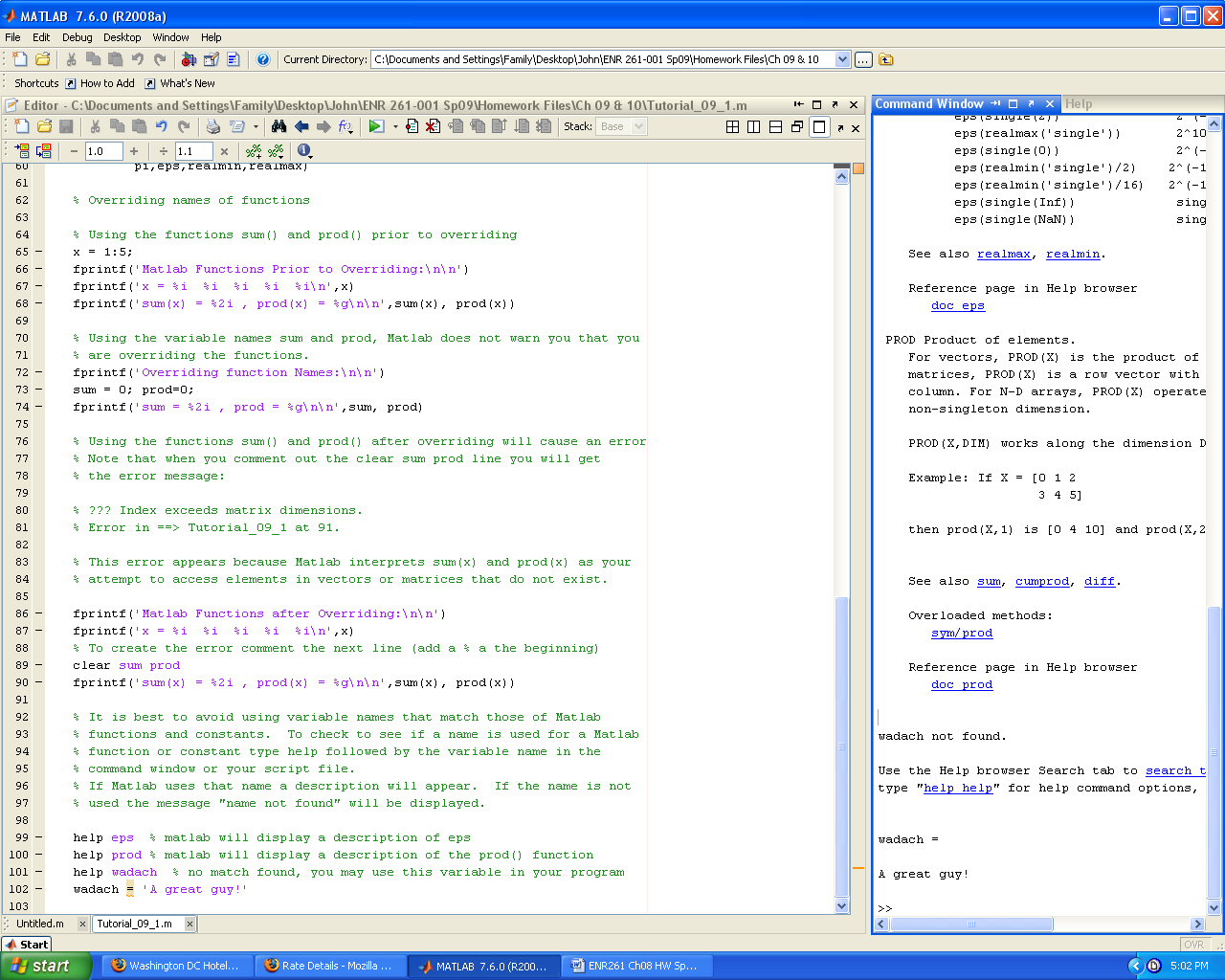
Required File Name: **Tutorial\_09\_1.m**

****

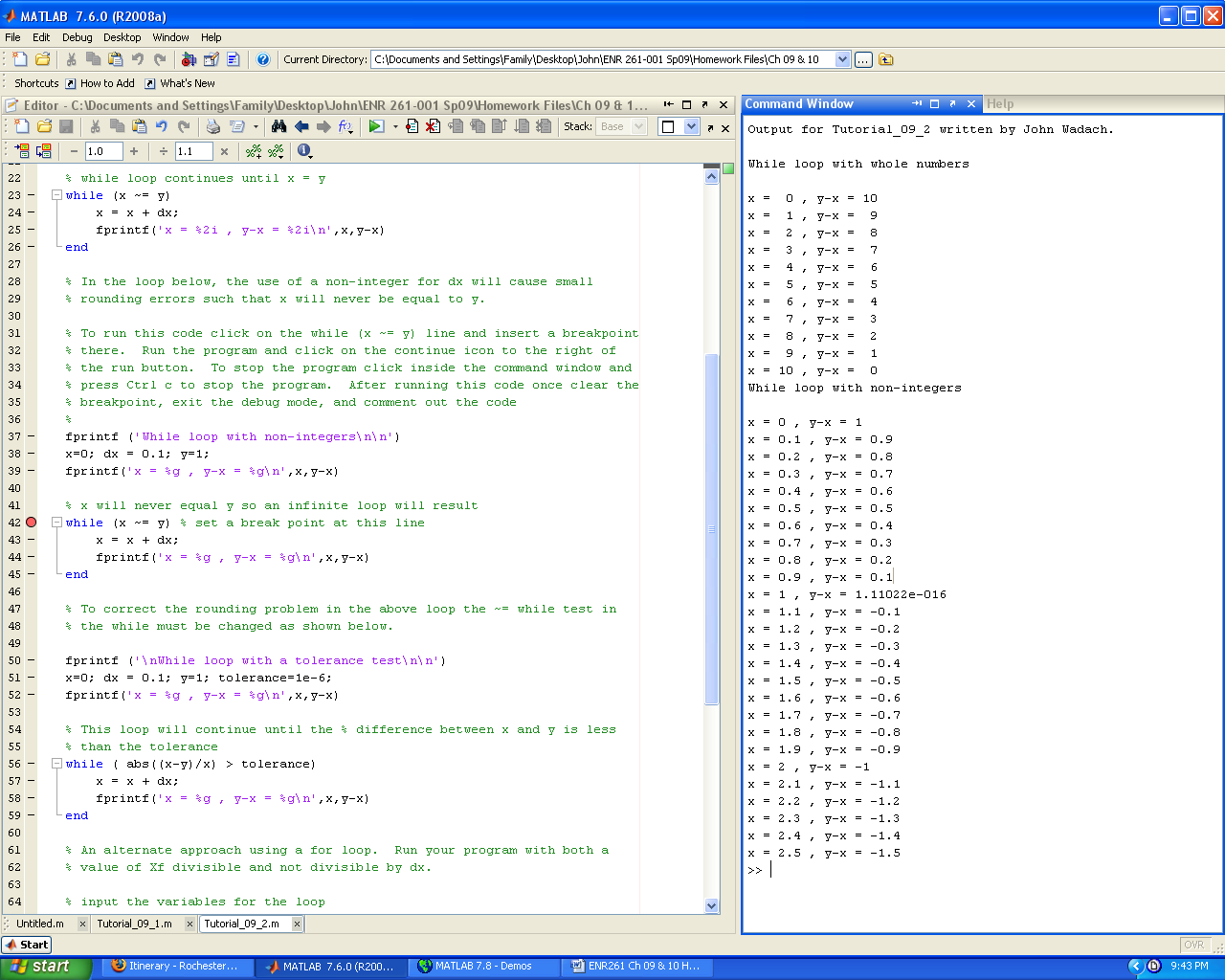
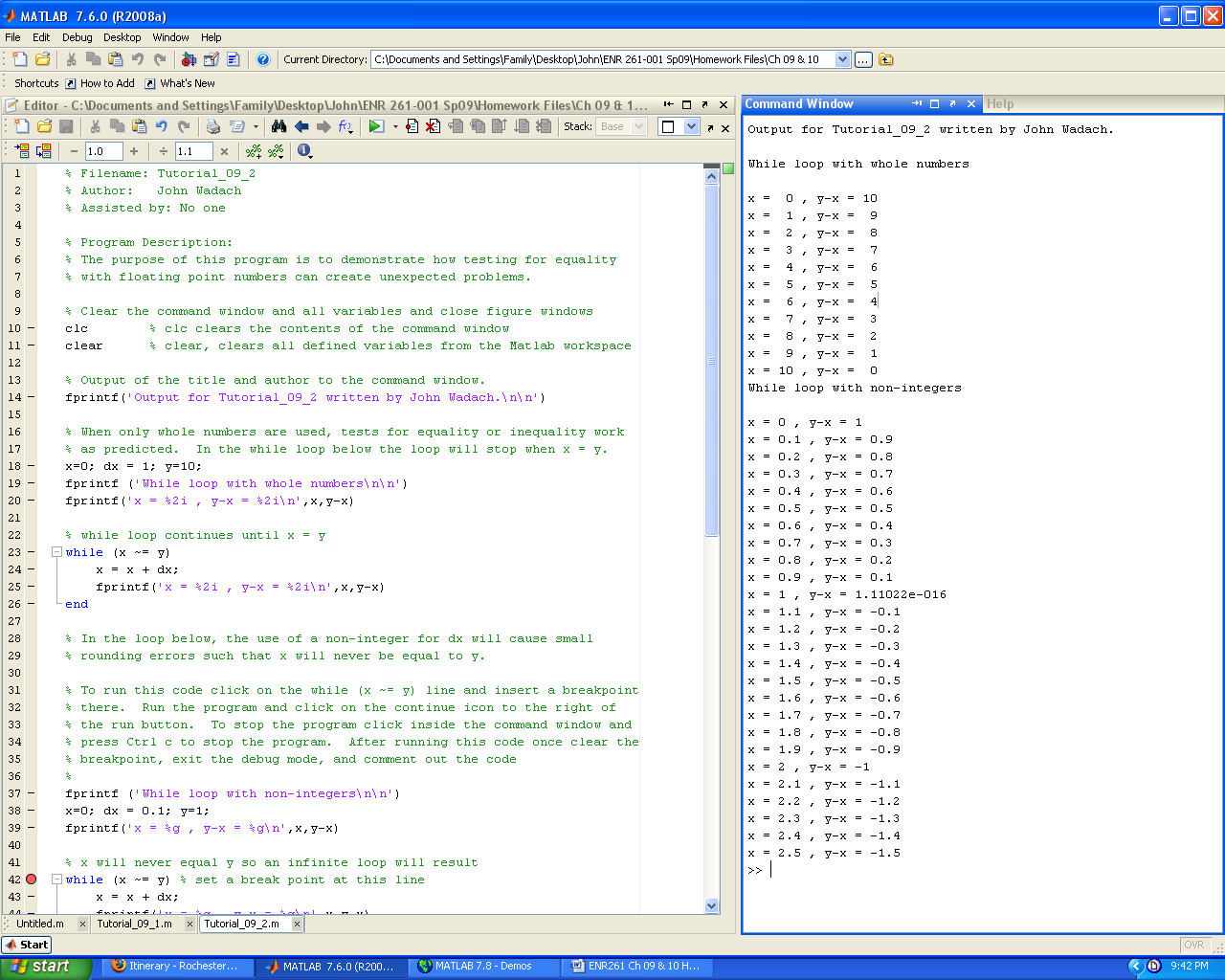
**Continued on the Next Page**

****

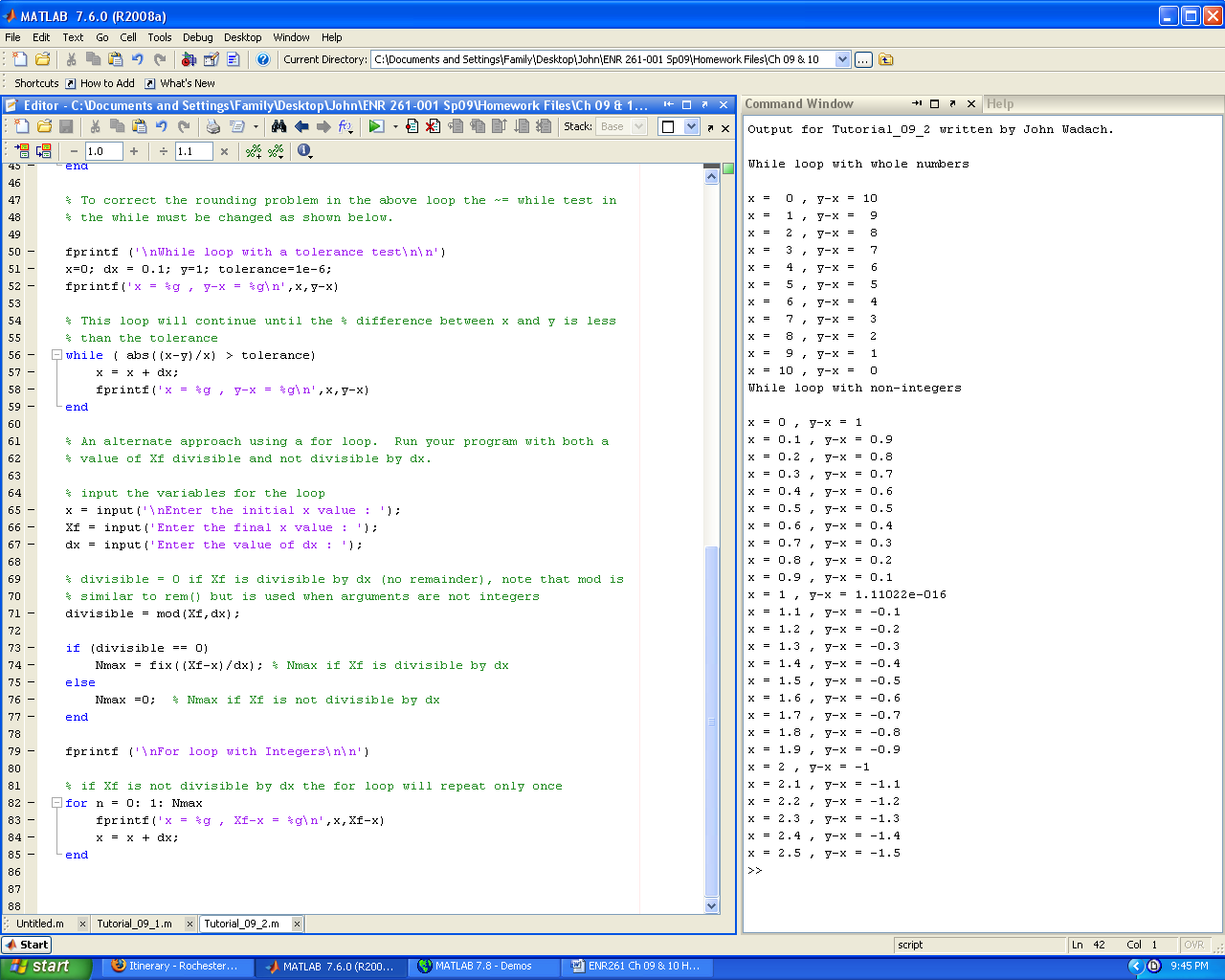
**Continued on the Next Page**

****

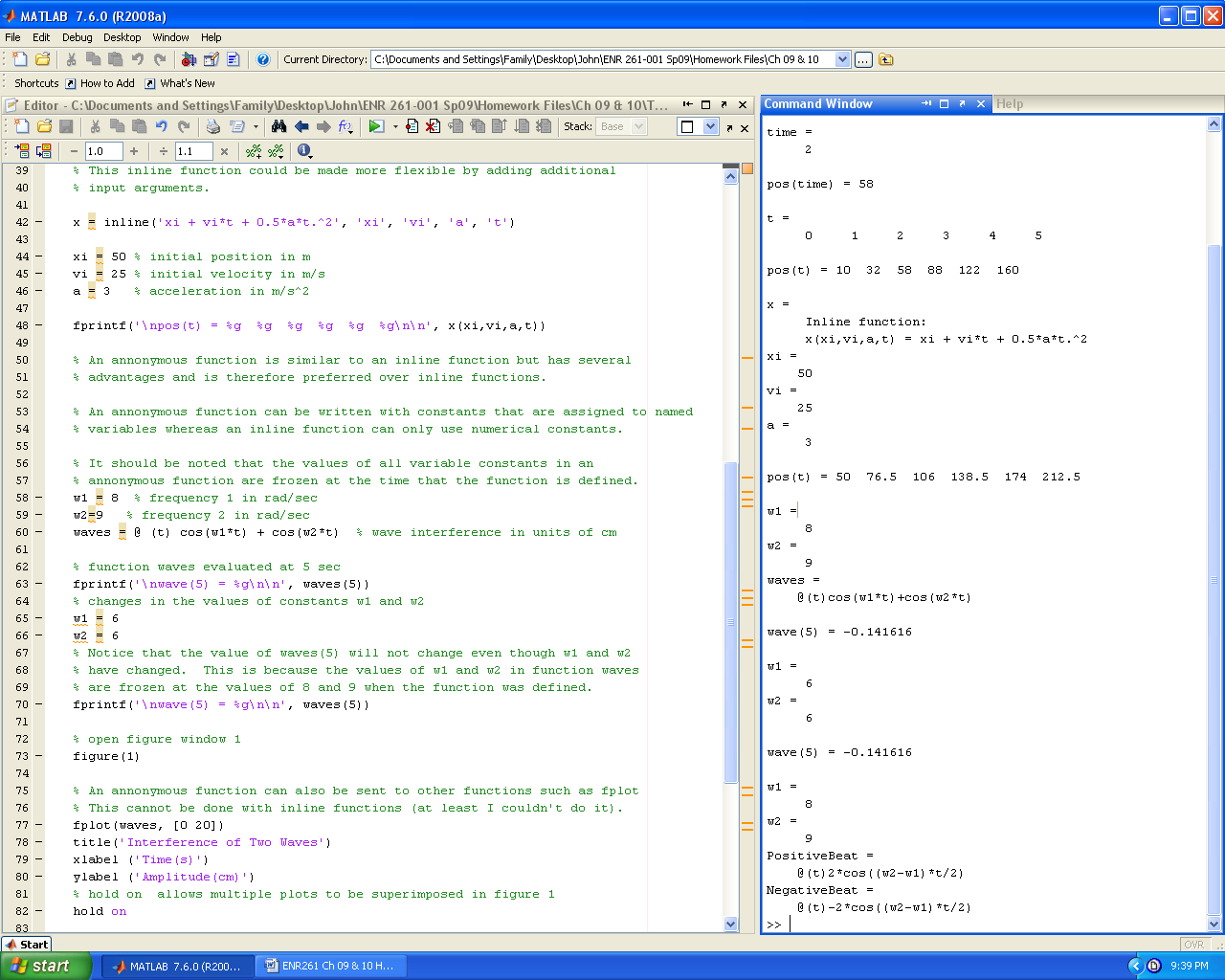
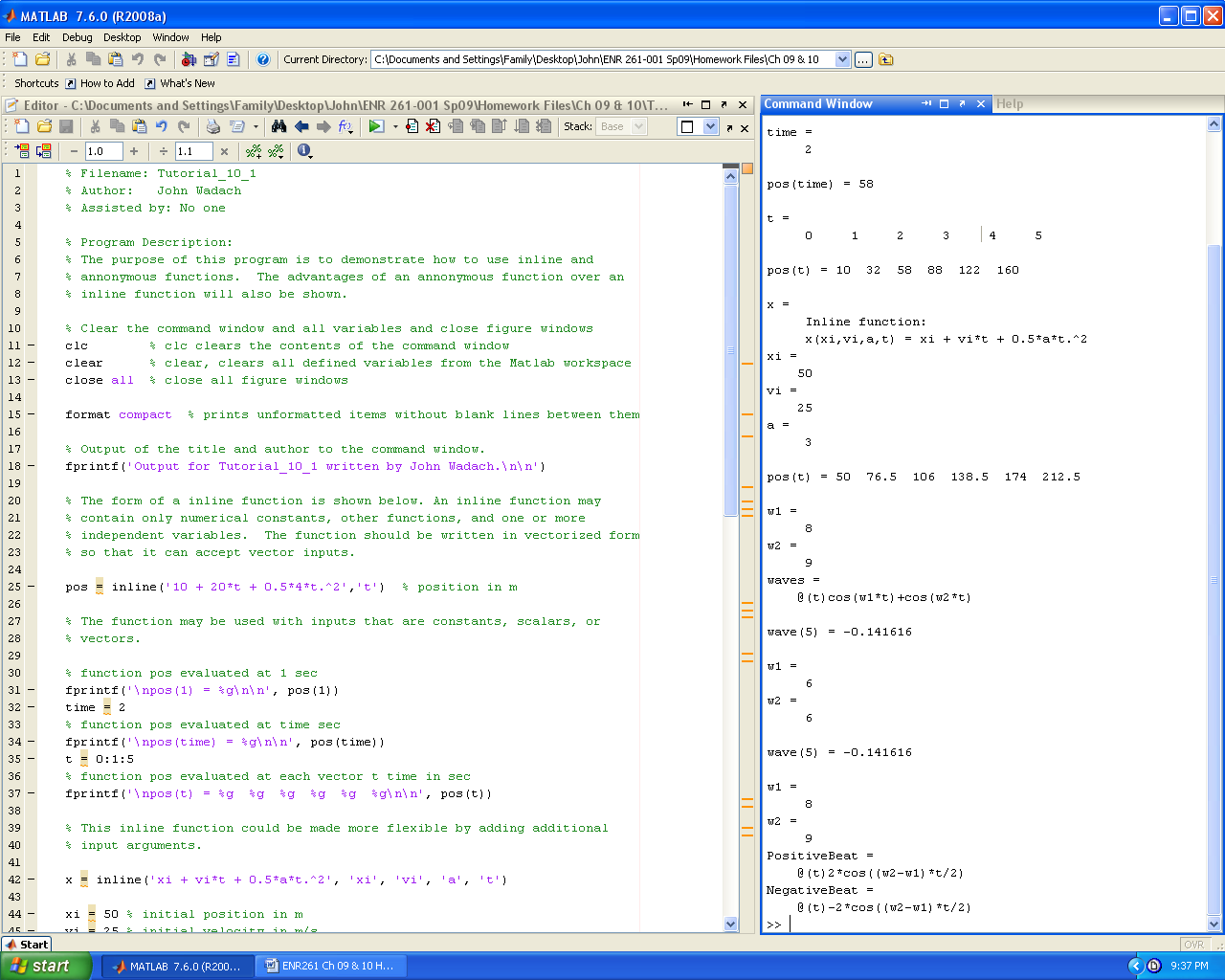
Required File Name: **Tutorial\_09\_2.m**

****

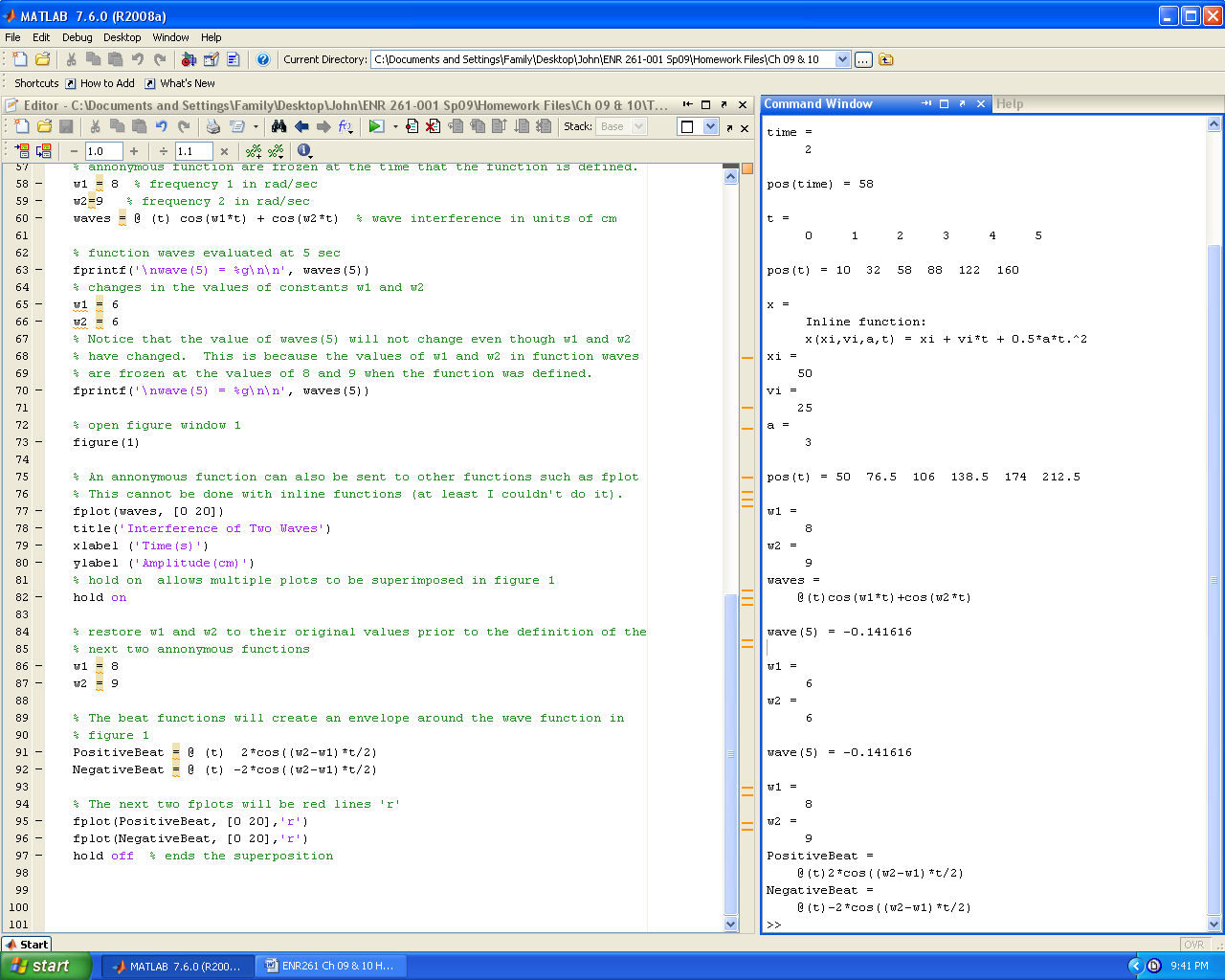
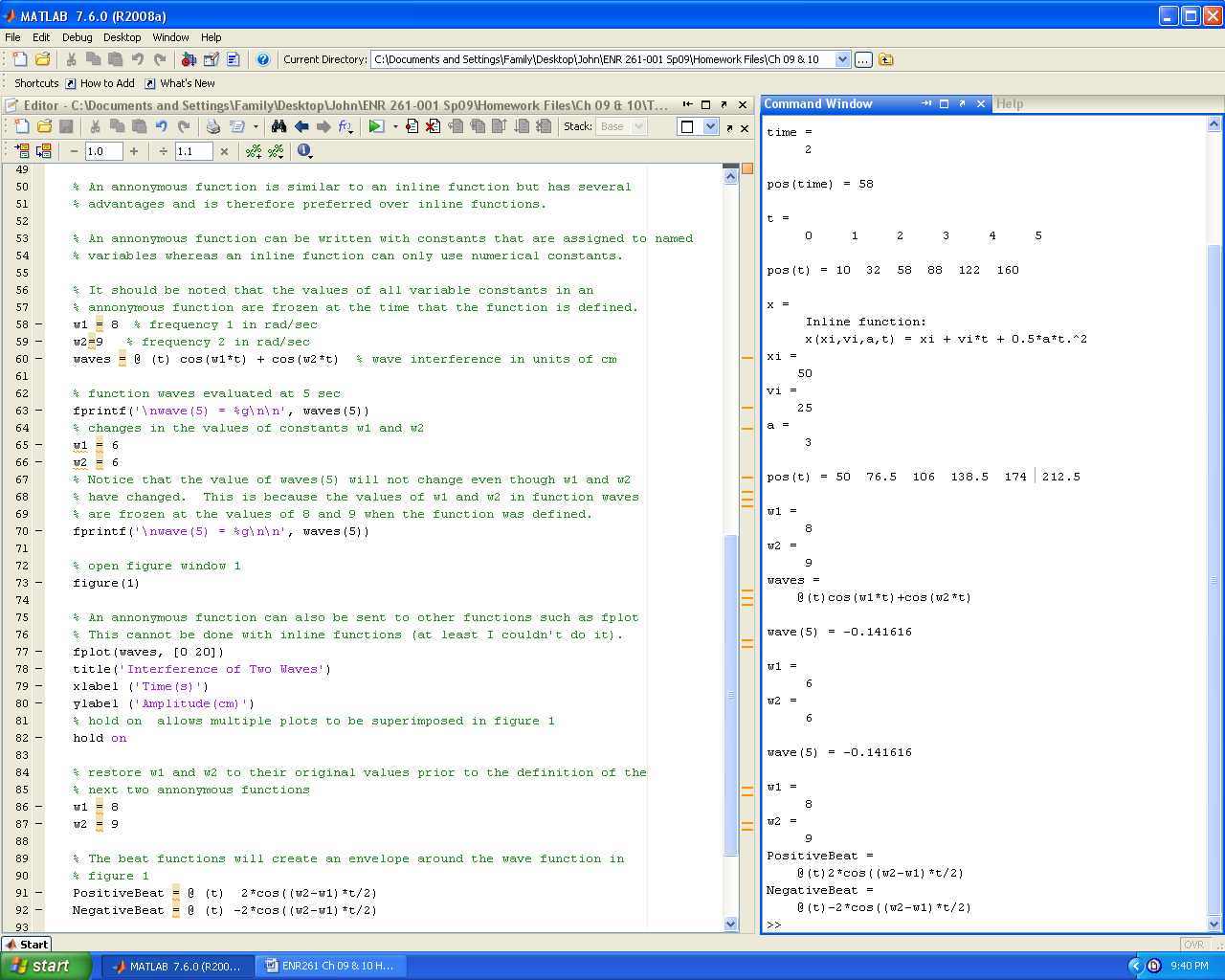
**Continued on the Next Page**

****

Required File Name: **Tutorial\_10\_1.m**

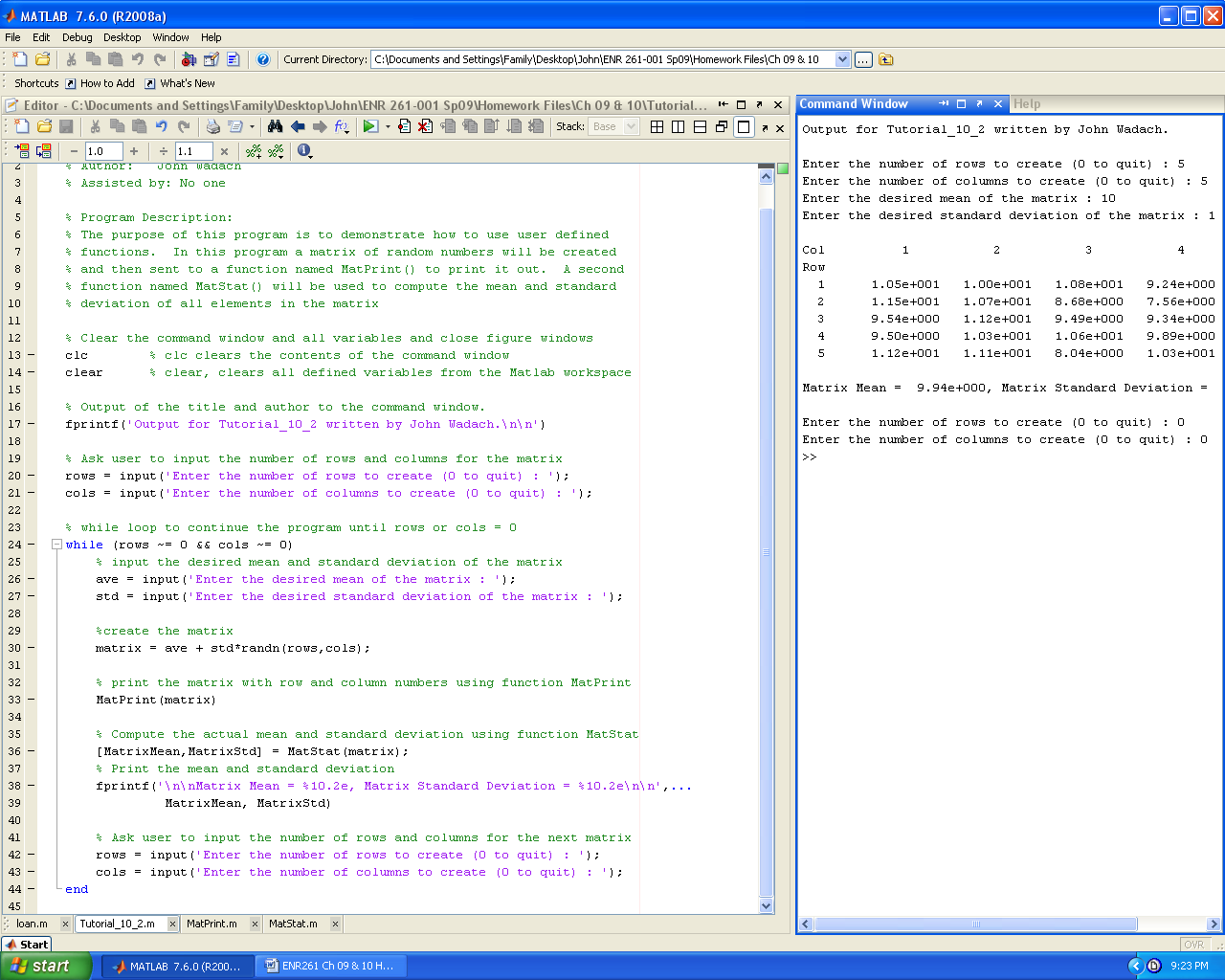
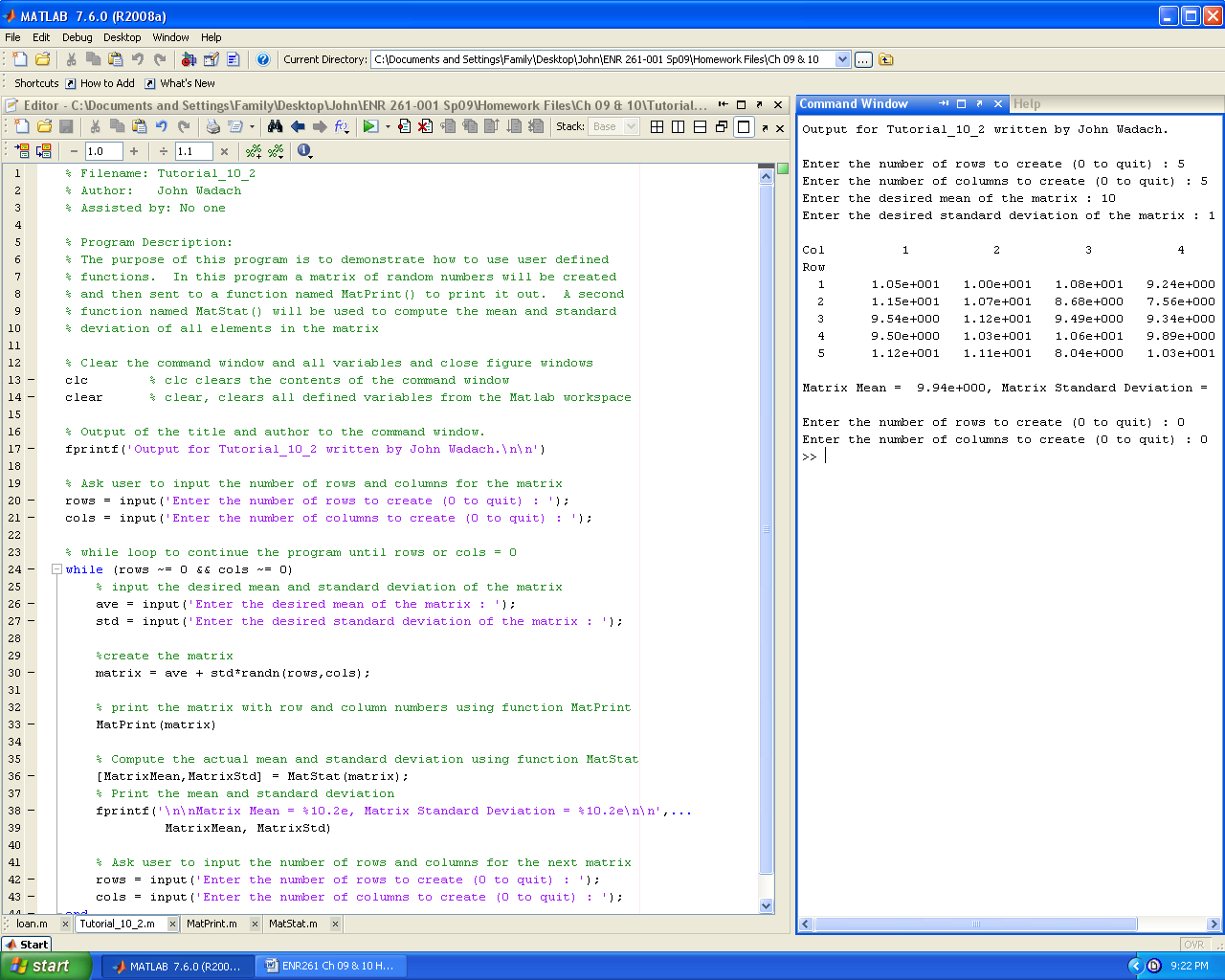
****

**Continued on the Next Page**

****

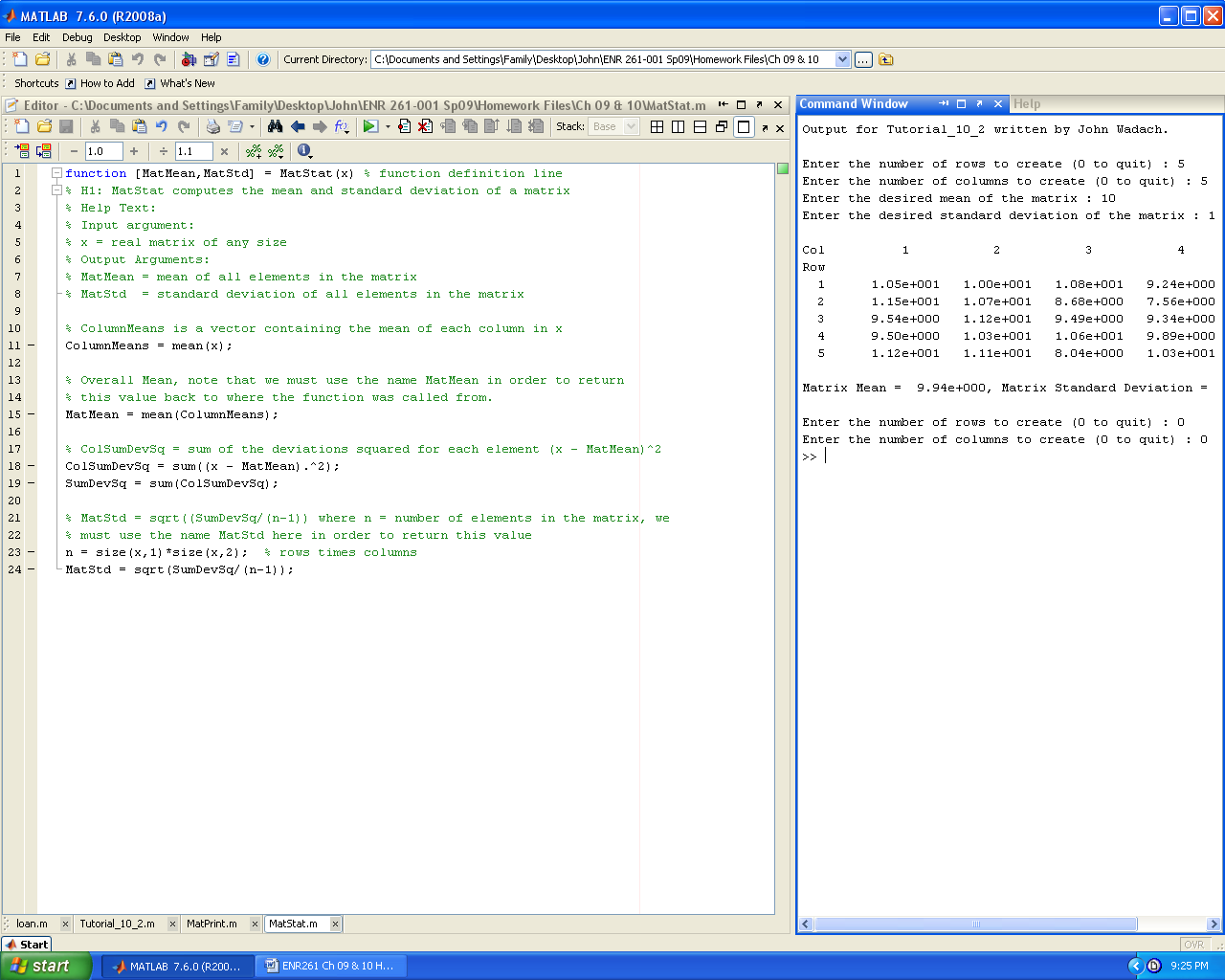
Required File Name: **Tutorial\_10\_2.m**

**(You will also have to have MatStat.m and MatPrint.m for this program to work)**

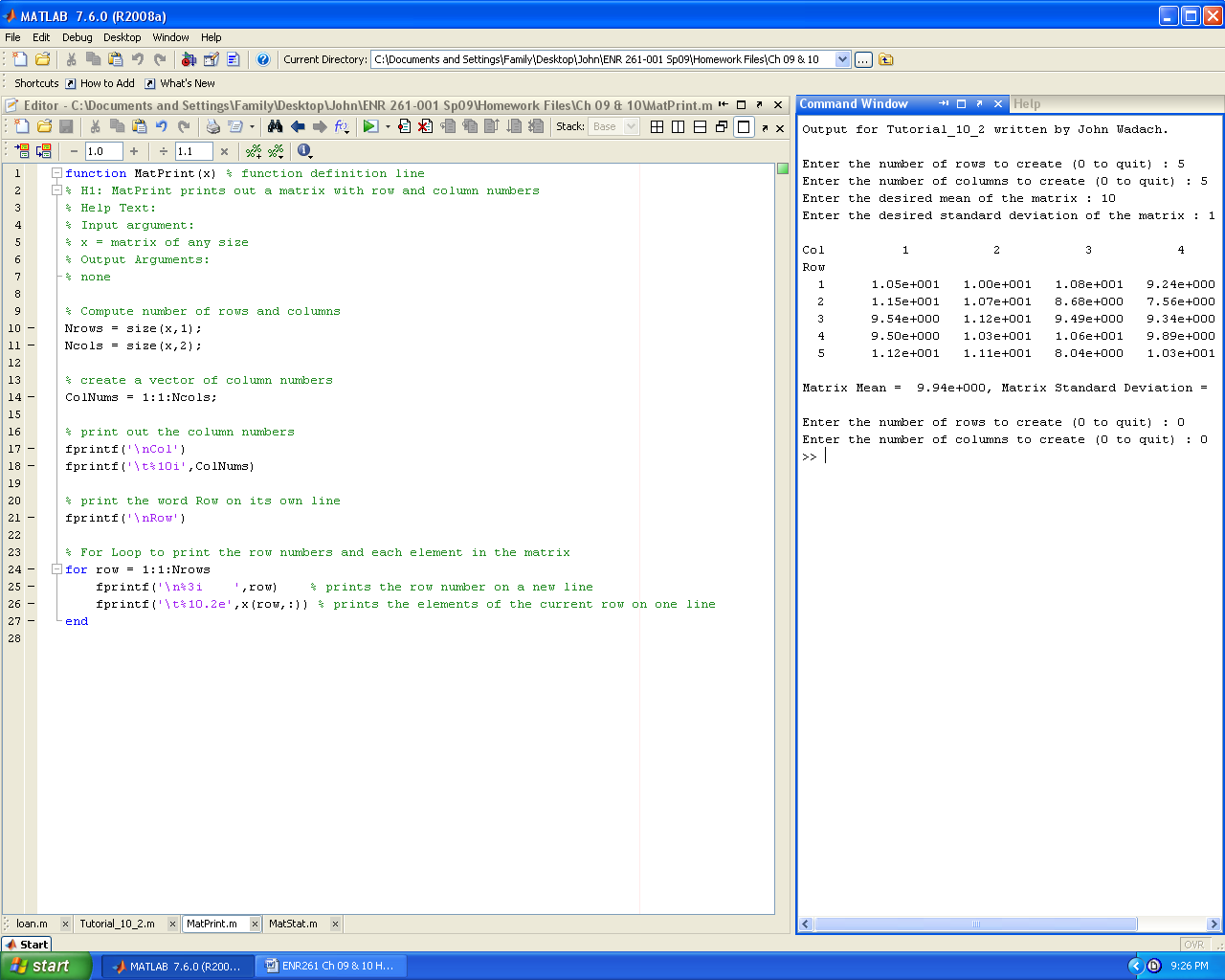
****

**Continued on the Next Page**

Required File Name: **MatStat.m** ( Type **lookfor MatStat** and **help MatStat** in the command window)

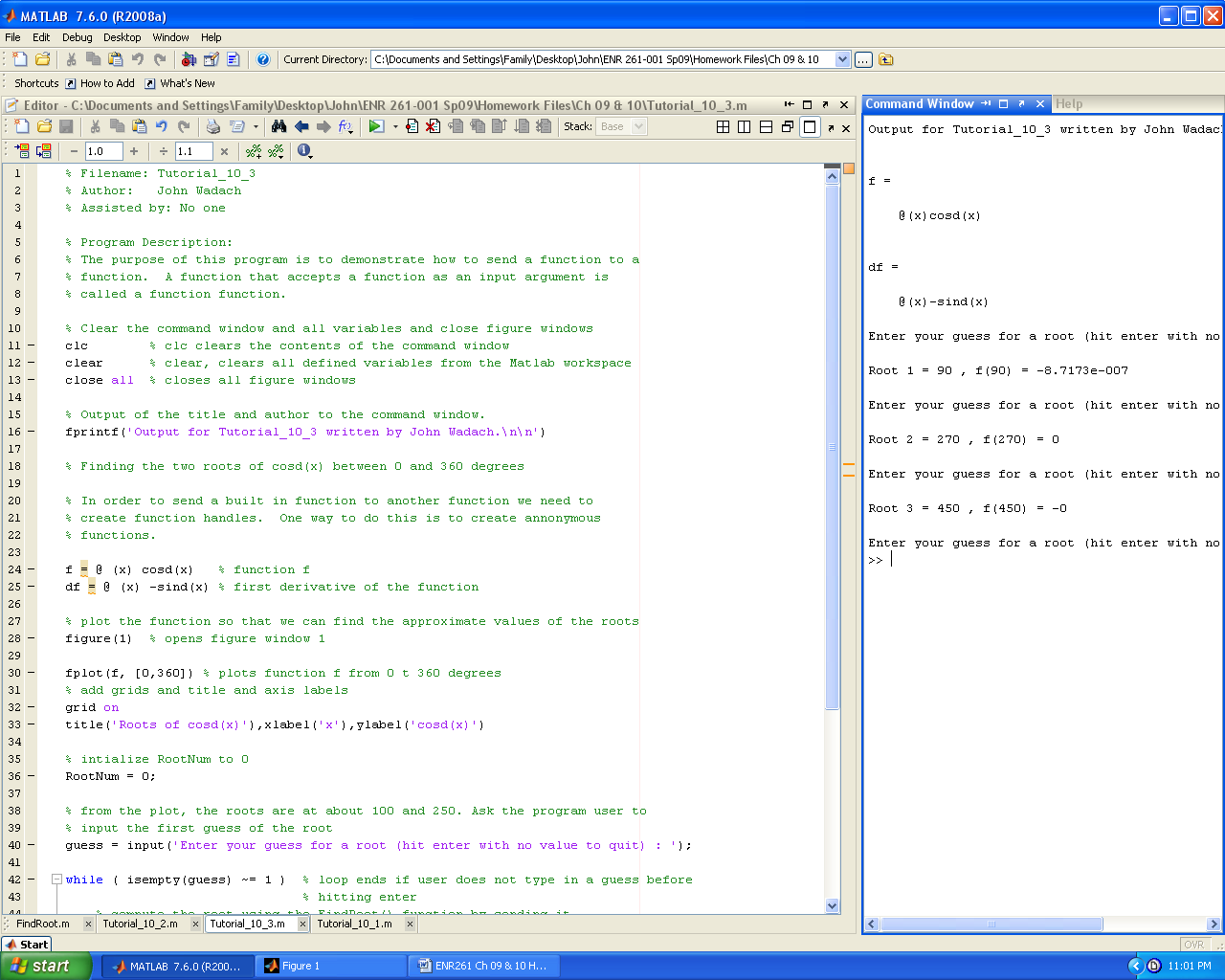
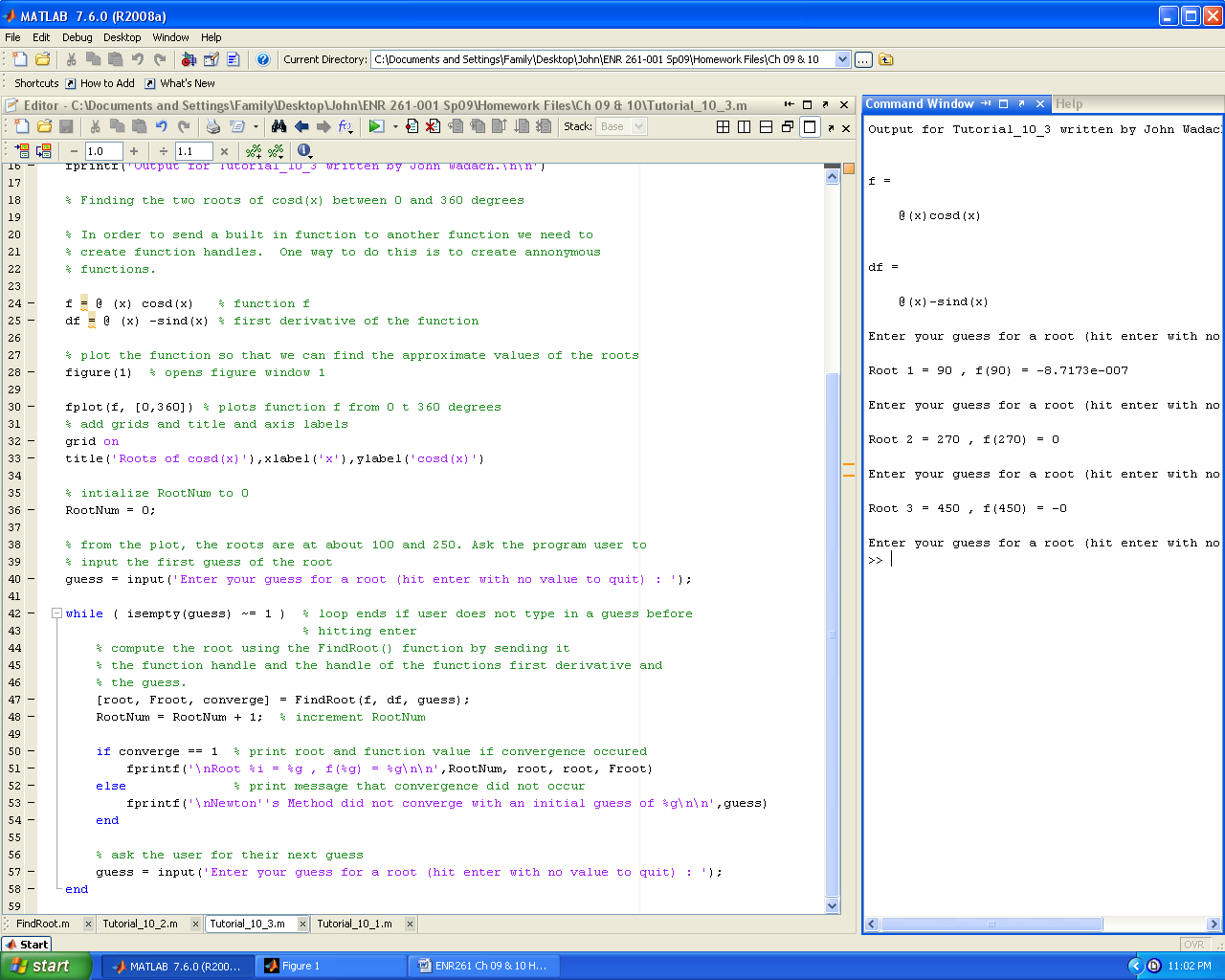
****

Required File Name: **MatPrint.m** ( Type **lookfor MatPrint** and **help MatPrint** in the command window)

****

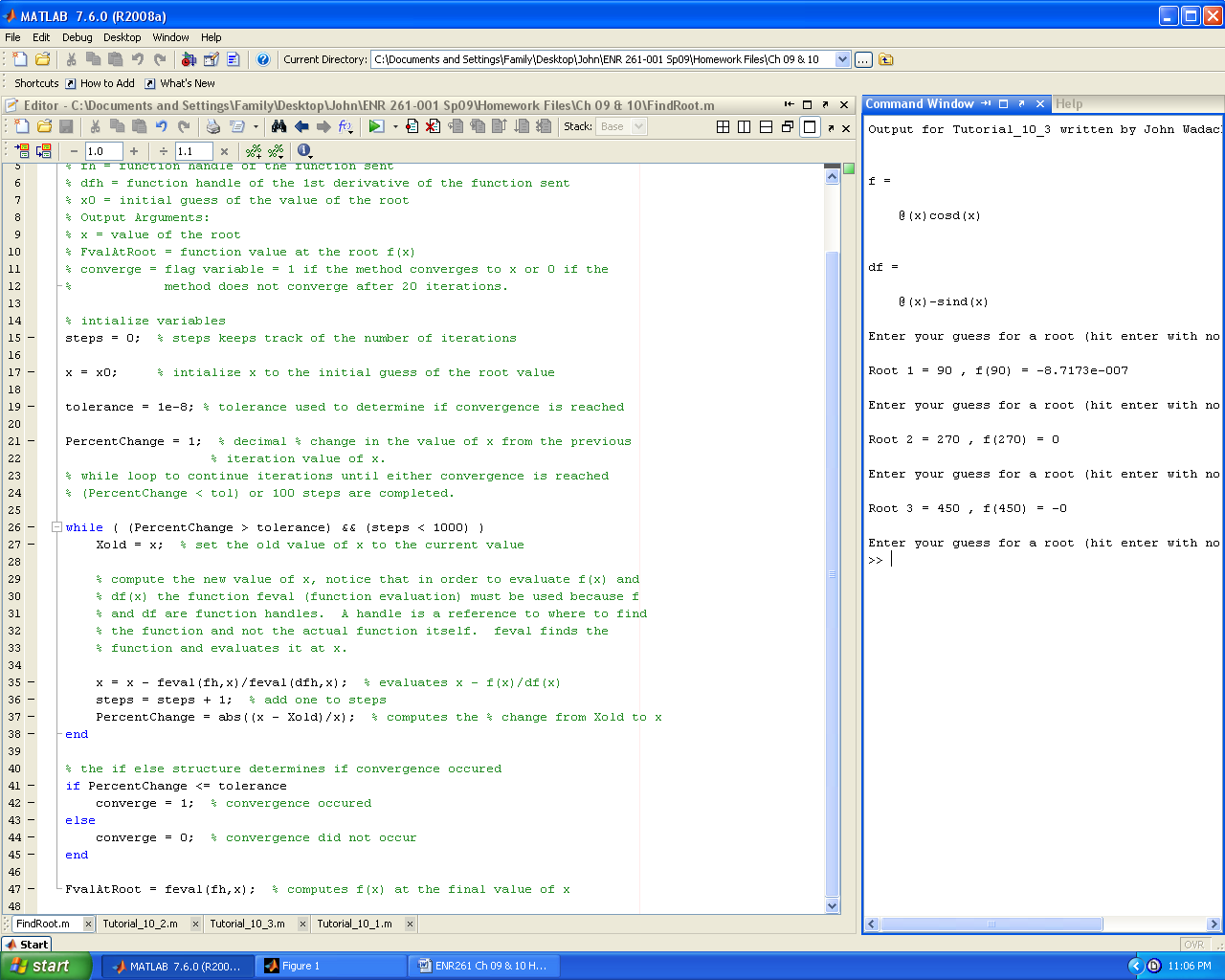
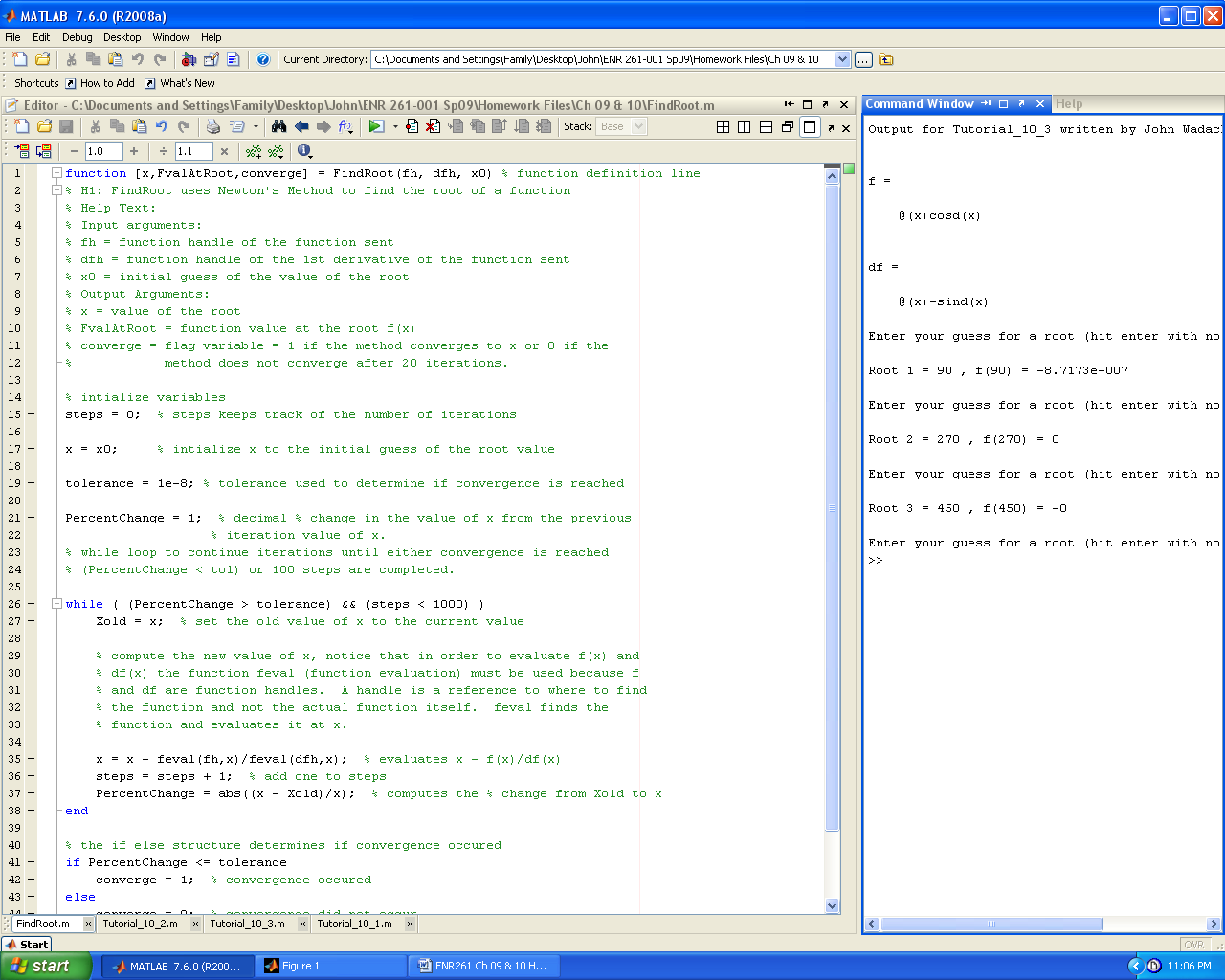
Required File Name: **Tutorial\_10\_3.m**

**(You will also have to create FindRoot.m for this program to work)**

** **

**Continued on the Next Page**

Required File Name: **RootFind.m** ( Type **lookfor RootFind** and **help RootFind** in the command window)

****

Required File Name: **Program\_10\_1.m**

Following the equation:

 vectorize your Taylor function so that a vector of x values can be sent to the function.

Use the following function definition and help text in your function.

function [Texp] = TaylorExp(x) % function definition line

% H1: Computes e^x using a Taylor Series for x >= 0

% Help Text:

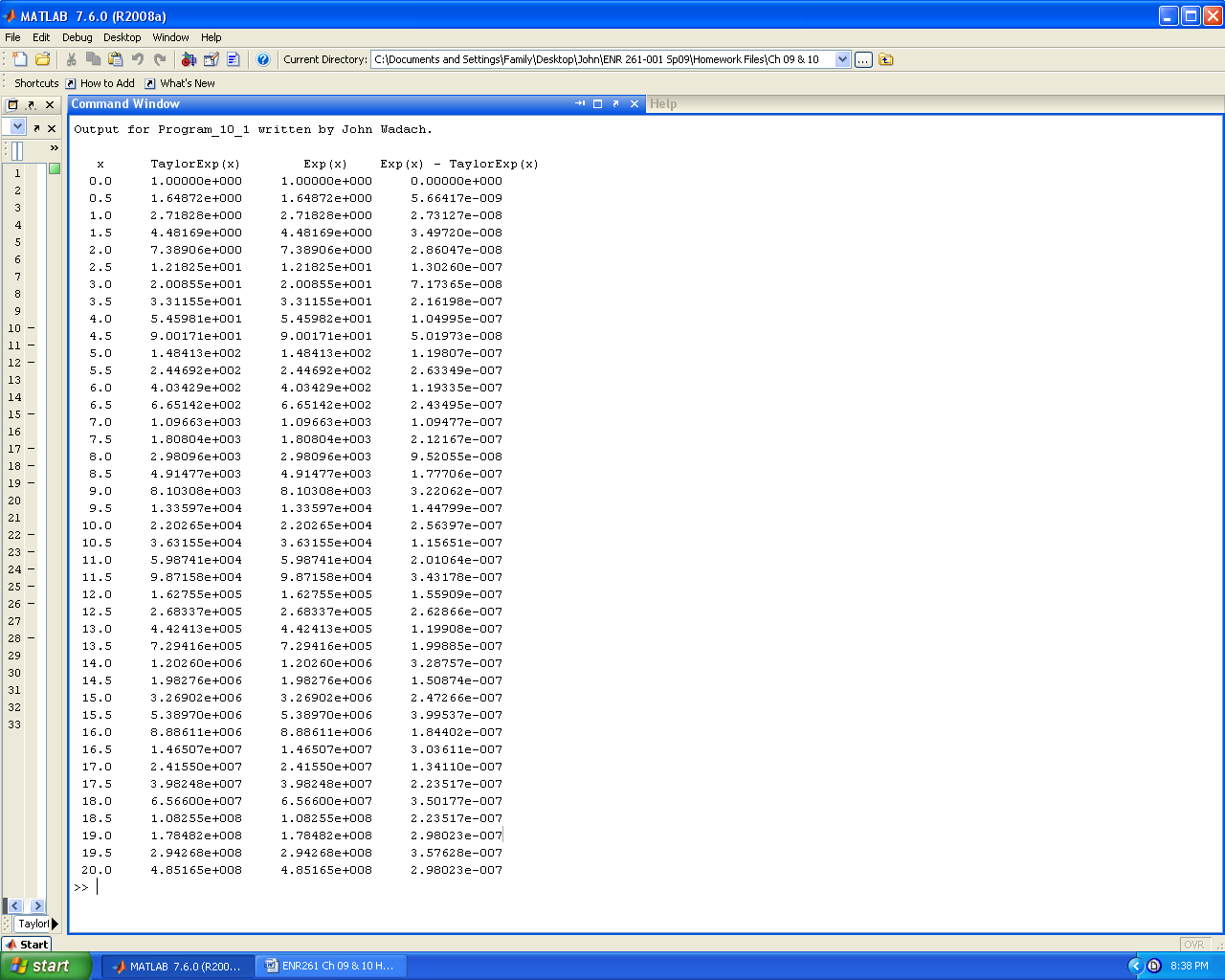
% Input argument:

% x = a vector of exponents, all values of x >= 0

% Output Argument:

% Texp = a vector of e^x values computed using a Taylor Series

In Program\_10\_1 create a vector of x values and use this vector along with the TaylorExp() and exp() functions to generate the table below as efficiently as possible.



Required File Name: **Program\_10\_2.m**

In this program use the user defined functions LabelPlot() and InteractivePlot() the function definitions are given below.

function LabelPlot(TitleText,Xtext,Ytext)

% H1: Adds a title, X and Y axis labels, and gridlines to a graph.

% Help Text:

% Input Arguments:

% TitleText = a string for the title of the graph.

% Xtext = a string for the X axis label of the graph.

% Ytext = a string for the Y axis label of the graph.

% Output Arguments:

% There are no input arguments.

function InteractivePlot(fh,xmin,xmax) % function definition line

% H1: Creates and formats a plot(x,f) using inputs from the command window.

% Help Text:

% 1000 values of x between xmin and xmax are used in the plot.

% The feval() function is used to evaluate the function at each value of x.

% The user inputs the line color, line style, and line width for the graph

% from the command window during execution of this function.

% Input arguments:

% fh = function handle of the function to be plotted

% xmin = minimum x value to plot

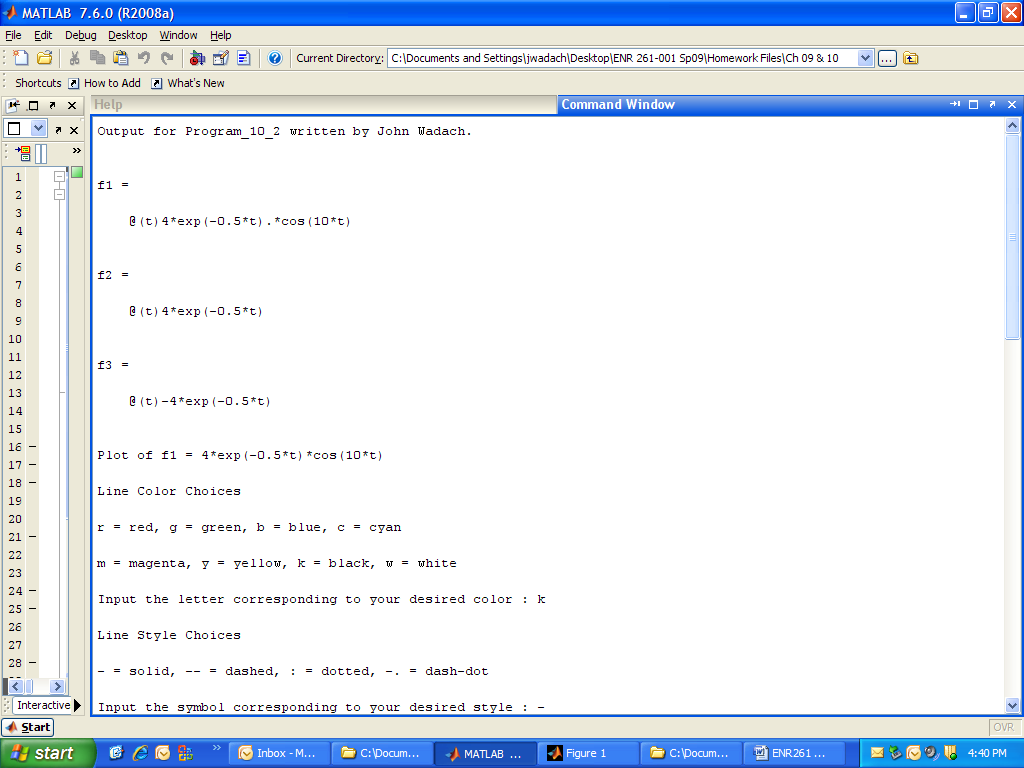
% xmax = maximum x value to plot

% Output Arguments:

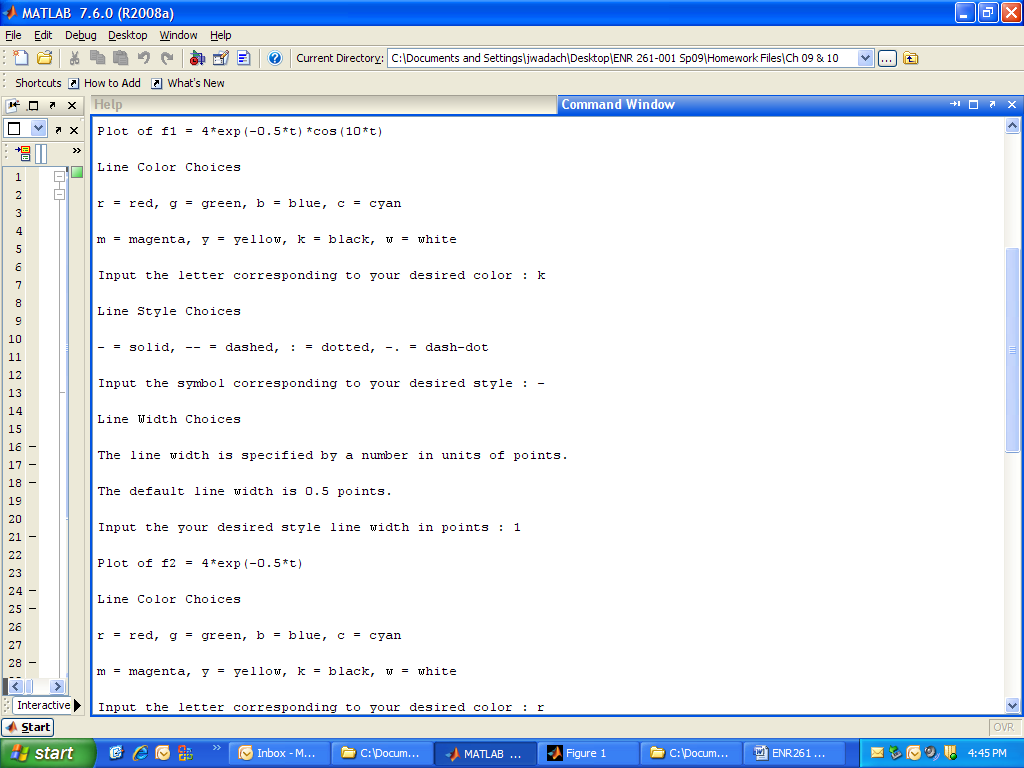
% None, this program does not return any values

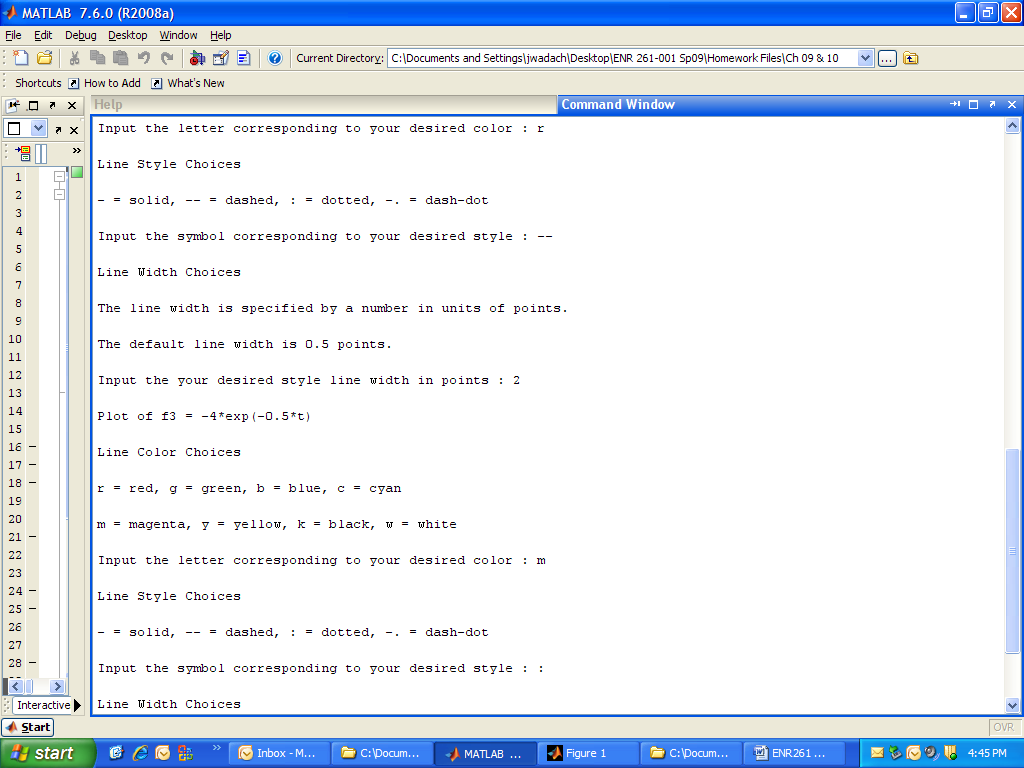
Program\_10\_2 defines the three anonymous functions and sends them to InteractivePlot() to create plots of the three functions on the same graph. The program also calls function LabelPlot() to add the title, gridlines, and axis labels to the graph. Finally, the program adds a legend to the graph.

A sample run of the program is shown below.

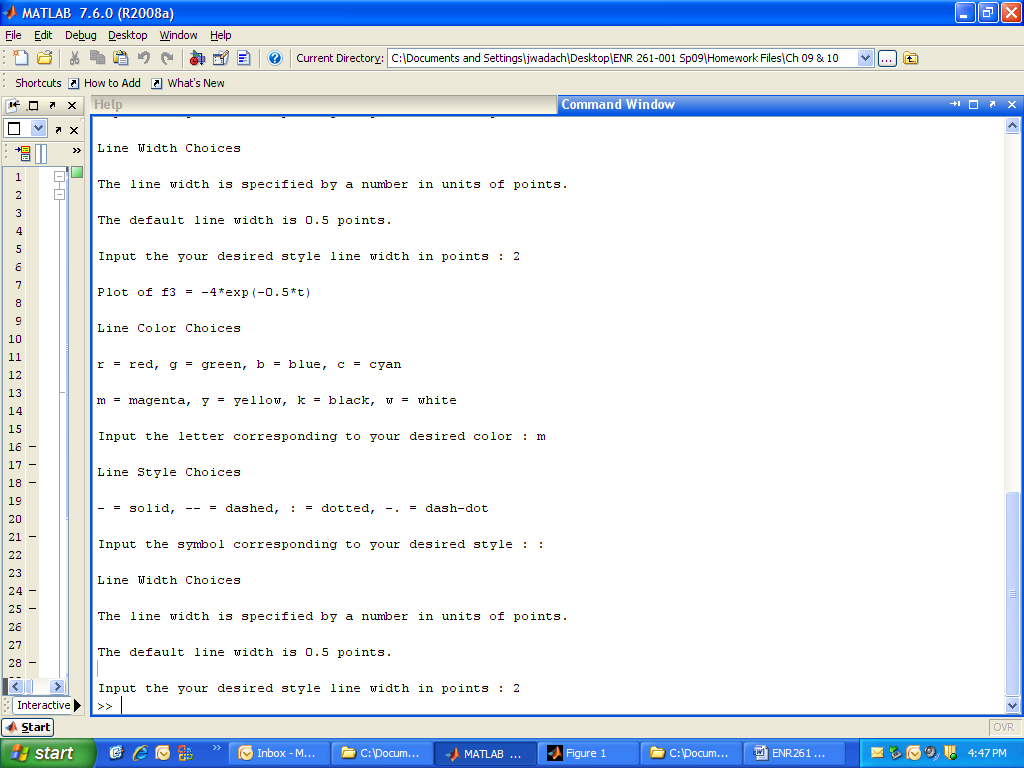


**Continued on the Next Page**





**Continued on the Next Page**





Required File Name: **Program\_10\_3.m**

In this program define the two vectors below.

% Given Data

x = [ 0 1 4 5 7 10 13 15 16 19 20];

y = [22 17 12 9 2 -6 -12 -20 -22 -28 -32];

Send this data to the user defined function LinearReg() to find the slope, y-intercept, and R^2 value.

function [m , b, Rsq] = LinearReg(x,y)

% H1: Computes the slope, y-intercept, and regression coefficient.

% Help Text:

% Linear regression is used to compute the values.

% Input Arguments:

% x = a vector of independent variables

% y = a vector of dependent variables

% Output Arguments:

% m = slope of the best fit line

% b = y-intercept of the best fit line

% Rsq = regression coefficient, 1 = perfect fit

Hint: See your ENR161 Spreadsheet Textbook for the formulas to use for m, b, and r^2.

Create the following output and graph in your program.

