# EGR 120 Due date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Introduction to Engineering

File: EGR120-MATLAB-C

 **Homework Assignment #6 (MATLAB Assignment C)**

**Reading Assignment:**

Read Chapter 15 in Engineering Fundamentals – An Introduction To Engineering, 5E by Moaveni.

PowerPoint: EGR120-MATLAB-C

**MATLAB Assignment:**

* **Warning**: Your assignments must be your own work. You can ask other students questions, but sharing files is cheating. If any evidence of copied files is discovered, all parties involved will receive grades of 0 and may be subject to further disciplinary action.
* For **all problems**, begin all MATLAB programs (scripts or .m files) with the following information:

 % John Doe (**your name**)

 % EGR 120

 % Homework Assignment #?, Problem ?

 % Filename: YourFileName.m

 % Instructions

* Use descriptive variable names
* Use ***format compact*** to reduce extra lines in the output.
* Print the program (script or .m file) and the results for each problem. If you post the results online, post both the program and the results.
1. (33 points) Write a MATLAB program to prompt the user to enter a value of x and y corresponding to the (x,y) coordinates of a point in the Cartesian coordinate system. The program should then display one of the following prompts:
* The point is on the x-axis
* The point is on the y-axis
* The point is in the first quadrant
* The point is in the second quadrant
* The point is in the third quadrant
* The point is in the fourth quadrant.

Test the program for all 6 cases. Turn in a printout of the program and a printout showing the 6 test cases. Include plenty of comments in your program.

(assignment continued on the next page)

1. (33 points) Write a MATLAB program to calculate the three angles in a triangle given the three sides using the law of cosines, but the program should first test to see if the three sides entered are valid.



A triangle with sides a, b, and c is valid if all of the following conditions are true:

A

a

C

b

c

## B

* a>0
* b>0
* c>0
* a<b+c
* b<a+c
* c<a+b
* Include comments in your program, including name, course, filename, description of the assigned problem, and explanations of program features. Also display a description of the program
* Prompt the user to enter the three sides of the triangle.
* Use a single if statement to test if all 6 conditions for a valid triangle are true.
* If the triangle is not valid, display an error message.
* If the triangle is valid, display the three sides as well as the three angles in degrees (include the unit ***degrees***).
* Test the program for ***6 invalid cases*** (one for each of the 6 conditions listed above).
* Test the program for ***3 valid cases***:

|  |  |  |  |
| --- | --- | --- | --- |
| Case | a | b | c |
| 1 | 20 | 15 | 10 |
|  2 | 5 | 12 | 13 |
| 3 | Any example you pick |

* Turn in a printout of the program and a printout of the output for the 9 cases above.

(assignment continued on the next page)

1. (34 points) Write a MATLAB program to find the roots of a second order polynomial equation of the form ax2 + bx + c = 0. Recall that the ***quadratic equation*** can be used to find the roots as follows:

Root1 = x1 = $\frac{-b+ \sqrt{b^{2}-4ac}}{2a}$

Root2 = x2 = $\frac{-b- \sqrt{b^{2}-4ac}}{2a}$

The roots may be real and distinct, repeated, or complex as described below.

|  |  |
| --- | --- |
| b2 > 4ac | Roots are real and distinct (different) |
| b2 = 4ac | Roots are repeated |
| b2 < 4ac | Roots are complex |

Specifically, the program should:

* Display a brief description of the program and the form for the quadratic equation.
* Prompt the user to enter values for a, b, and c.
* Display an error message if a, b, or c are equal to zero.
* Test to see if the roots are real & distinct, repeated, or complex. Use an efficient if structure to display the type of roots.
* Use the equations provided above to find the two roots.
* Note: Do not use fprintf in this case. It is easiest not to use special formatting using fprintf with complex numbers. If you simply calculate the roots (with no ;), MATLAB will display complex roots correctly for each of the cases.
* Run the program for the three examples below and for one case where a, b, or c are equal to zero.
* Turn in a printout of the program and a printout showing the ***7 examples***:
	+ The 3 examples below
	+ 3 more examples (one with real roots, one with repeated roots, one with complex roots)
	+ 1 example where a, b, or c is equal to zero
* Include plenty of comments in your program.
* Note that there is a function called **roots** in MATLAB which can be used to find roots of polynomials. You cannot use this function or any similar function in this assignment. Use the equations provided.

|  |  |  |
| --- | --- | --- |
| Example 1: x2 + 30x + 200 = 0Program Output:***Roots are real and distinct:******x1 =*** ***-10******x2 =*** ***-20*** | Example 2: x2 + 12x + 36 = 0Program Output:***Repeated Roots:*** ***x1 =*** ***-6******x2 =*** ***-6*** | Example 3: x2 + 8x + 25 = 0Program Output:***Complex Roots:******x1 =*** ***-4.0000 + 3.0000i******x2 =*** ***-4.0000 - 3.0000i*** |