A05 · MATLAB Scripts

Introduction

Assignment Goals

This assignment will give you experience working with matrices and will allow you to create and run MATLAB scripts with engineering contexts. You will gain programming experience with MATLAB plotting statements and then you will apply those skills to an engineering context. You will do the same for displaying information as text and data as plots, which you will also format for technical presentation.

Successful Completion

This assignment has four (**4**) problems. The deliverables list contains everything you are expected to submit.

Submit all problems to the Gradescope online assignment **A05 – All Problems**		
Problem	**Type**	**Deliverables**
Problem 1: MATLAB Skills – Matrix Manipulation	Individual	☐ Requested code snippets
Problem 2: Programming Standards	Individual	☐ A05Q2_ARcoating_*login*.m
Problem 3: MATLAB Skills - Plots	Individual	☐ A05Q3_plots_*login*.m ☐ Image files: ○ A05Q3_figure1_*login*.png ○ A05Q3_figure2_*login*.png ○ A05Q3_figure3_*login*.png
Problem 4: Gate and Interconnect Delay	Individual	☐ A05Q4_delays_login.m ☐ Supporting files: ○ Data file loaded into m-file ○ A05Q4_figure_*login*.png

1. Read *Notes Before You Start*, on **Page 2**.
2. Read the programming standards document included in the assignment files. Be familiar with this document; all programs in the class are expected to follow these standards.
3. Read each problem carefully. You are responsible for following all instructions within each problem.
4. Complete the problems using the problem-specific m-file templates when a template is provided in the assignment download. Replace *template* in the filename with your Purdue Career Account login.

   Example: `A05Prob1_context_template.m` will be renamed `A05Prob1_context_pboilerm.m` for a student whose Career Account login is pboilerm.
5. When your work is complete, confirm your deliverables are properly submitted to Gradescope.

Learning Objectives & Grading

This course uses learning objectives (LOs) to assess your work. You can find a list of the course LOs on Brightspace (Content > Key Course Info > Learning Objectives).

Review the assignment grading for each problem in this assignment, which starts on **Page 12**. This outlines how your work will be graded for each problem.

Notes Before You Start

Helpful MATLAB Commands

Learn about the following built-in MATLAB commands, which might be useful in your solutions:

`sum, sgtitle, plot, title, xlabel, ylabel, legend, close`

M-file vs Live Code in MATLAB

You must use m-files for this course. You can confirm your submission is the correct type if it has the ***.m** extension. Live files with the .mlx extension will not be accepted.

Document, Test, Debug, and Finalize Your Code

- Comment your code **while you are coding, not afterwards**. It is easy to forget what each line of code represents if you delay commenting and waiting until the end to add comments increases the time you will spend on commenting.
- Re-save, run, and debug your code often, preferably after each new line or closely related 2-5 new lines of code are added. This allows you to identify the true location of problems more easily. MATLAB identifies the first line of code that fails, but the actual error could be on any previous line.
- Suppress printing of code that is functioning properly. Only formatted displays should be printed in the Command Window once your code is functional.
- Ensure your script will run when the Workspace is empty. A properly functioning script should contain all the necessary commands within the script itself. Be careful if you test commands or store variables in the Workspace that are not created from the script. To test your final code, clear the Workspace completely and then run the script from the command prompt. Only variables created by your script should remain in the Workspace and no errors appear in the Command Window.

Creating Plots of Data

- When analyzing bivariate (two variables) data, you must determine which is the independent variable and which is the dependent variable.
- A common way to phrase a request for a plot is to say, "Plot variable 1 versus variable 2." Variable 1 refers to the y-axis variable; variable 2 refers to the x-axis variable.
- Be sure to use the proper line and/or marker style for this class.

Creating Plots of Models

- When plotting models, use lines with no data markers. The points used to generate the plot are selected for convenience and do not refer to actual data.
- You can present a model with its raw data on the same plot. The model is a line with no data markers. The raw data are data markers and no connecting line.

Testing and Debugging Plot Code

- **Always close all figure windows before re-running your code.** Otherwise, your code will add or remove things from the existing plot displays. This can appear randomly in ways that may or may not reflect the presence of any coding problems.

Gradescope

You will submit all your deliverables to Gradescope for grading.

Gradescope has several types of assignments, and the submission process varies depending on the assignment type. The most commonly used assignment types in First Year Engineering are:

- "programming assignment" where you to submit files, usually code and supporting files,
- "online assignment" where you submit answers and/or files within a Gradescope interface, and
- "homework assignment" where you submit an entire assignment as one PDF file.

This assignment has one online assignment named **A05 – All Problems**.

Need to access Gradescope?

1. Log into Brightspace and open your ENGR 132 course.
2. Click **Content** from the black menu ribbon at the top of the page.
3. Click **Gradescope** from the Table of Contents in the left sidebar.
4. Click the top item, which is a link that will open Gradescope within Brightspace.
5. Select the assignment you are ready to submit.

Be careful to name your deliverables correctly before you submit them. Submit all the files requested. Do not include files not listed in the deliverables.

Opening Gradescope through Brightspace will auto-enroll you in the Gradescope course for your section.

Need help with Gradescope?

Navigate to the same Brightspace location as above and view the links and documents in Gradescope Help.

Problem Generator File (A05_Skills.p)

In the assignment folder, you will see a file named **A05_skills.p**. This is a MATLAB function file that generates problem information for each skill problem in this assignment.

To use this file you must do the following:

1. Open MATLAB and identify your current folder.
2. Download **A05_skills.p** into your current MATAB folder (this can be on your personal computer or in your own MATLAB Drive folder).
3. Once the .p file is in your current folder, you can use it like a built-in function. The file requires two inputs.
    a. Input 1: Your 8-digit PUID number (you can leave off the leading 00s)
    b. Input 2: the problem number

    Example of what you will enter at the Command Line in the Command Window:

    ```
    >> A05_skills(12345678,1)
    ```
4. When you run this command, you will get information displayed to the Command Window, to a figure window that opens separately, or both. No information will be stored in your Workspace when you use the **A05_skills.p** function.
5. **You cannot open the p-code file**; you can only run it with the specified inputs.

Need help using the problem generator file? Check out the A00 Activity from Class 1B here to see fully-worked examples.

Problem 1: MATLAB Skills – Matrix Manipulation

Introduction

This problem allows you to practice array indexing and concatenation of arrays. You will perform all of the commands in the Command Window.

Problem

This problem has six (6) steps. You will start by creating a matrix and will then use array indexing and concatenation to revise the matrix. You need MATLAB with **A05_skills.p** in your current folder.

Instructions

1. Type this command into the MATLAB Command Window prompt:

   ```
   >> A05_skills(PUID, 1)
   ```

   Where you replace PUID with your 8-digit Purdue University ID number (leave off the leading 00).
2. This problem displays information for all six steps at once. Read the written instructions that appear in the Command Window.
3. Write the required commands. Work through the steps in order. Where applicable, assign the result to the variable given. You will only submit the commands to Gradescope, not any numeric results. You should never hardcode values from the matrix into a command.
   a. You may find it useful to save your commands to Gradescope after each step.
4. Submit your work in Gradescope:
   a. Open Gradescope > **A05 – All Problems** and find the set of boxes that belong to **Q1**. Enter the required information along with your answer:
      - ☐ **Function call**. Copy the command that you entered at the command prompt to call the function and paste the full command into this box. Be sure your PUID is included.
      - ☐ **Instruction text**. Copy the instruction text that is displayed in the Command Window. Paste it into this box. Include all text provided.
      - ☐ **Solutions**. Enter your solution(s). Follow any additional instructions.
   b. When you have entered all the required information for the question, click the **Save Answer** button.

Problem 2: Anti-reflective Coating

Introduction

Good programming standards allow you to keep your code well-organized, easy to interpret, and more flexible. Practice making a provided script follow good programming standards.

Problem

Nighttime drivers who wear eyeglasses can experience glare from the light of other vehicles. A chemical engineer has developed an anti-reflective coating for eyeglasses that improves the issue of glare by increasing the amount of light transmitted through the lenses of eyeglasses. The engineer started a script to perform the calculations necessary to determine the intensity of light passing through a coated lens and a non-coated lens. This information will allow the engineer to understand how well the coating works.

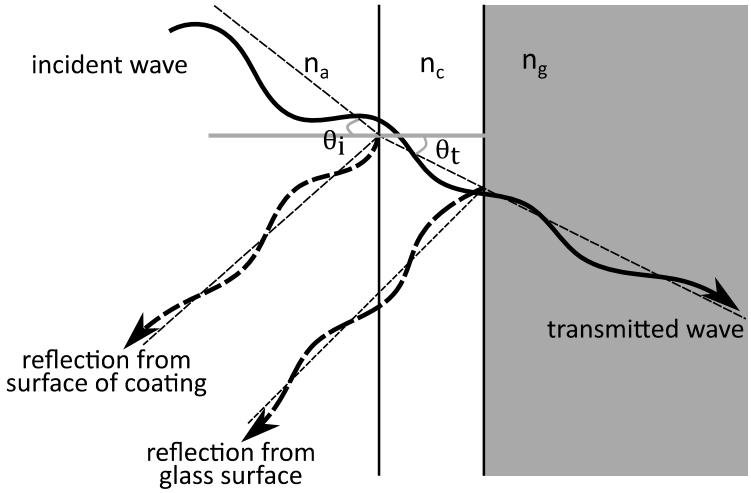
You have been given an m-file named **A05Q2_ARcoating_template.m** that contains the engineer's script in a standard ENGR 132 template. The script does not follow good programming standards, but you can assume that all calculations within script are correct.

You must reformat the script to make it follow good programming standards. You may need to create new variables or change variable names within calculations. Do not change anything within the `fprintf` statements.

Read the following context to understand the values and equations within the script.

Context

The schematic below shows an incident wave travelling through air, with majority of the wave being transmitted through the lens and part of it being reflected through two surfaces, the surface of the reflective coating and the surface of the glass. The chemical engineer wants to calculate the transmitted intensity through the eyeglasses with the coating ($T_{coating}$) and compare it to the light transmitted without the coating ($T_{nocoating}$). The higher the transmitted intensity, the lower the glare on the eyeglasses with a transmitted intensity of 1 meaning that all the light is transmitted and there is no glare. For simplicity, we are assuming the angle of the transmitted wave, θ_t, to be unchanged between the coating and glass.



The chemical engineer uses the equations below to calculate the percentage of transmitted light with and without the coating.

With coating

$$R_{coating} = \left(\frac{n_c \cos\theta_i - n_a \cos\theta_t}{n_c \cos\theta_i + n_a \cos\theta_t} - \frac{n_g \cos\theta_i - n_c \cos\theta_t}{n_g \cos\theta_i + n_c \cos\theta_t}\right)^2$$

$$T_{coating} = 1 - R_{coating}$$

Without coating

$$R_{nocoating} = \left(\frac{n_g \cos\theta_i - n_a \cos\theta_t}{n_g \cos\theta_i + n_a \cos\theta_t}\right)^2$$

$$T_{nocoating} = 1 - R_{nocoating}$$

Where,

n_a is the refractive index of air (unitless)

n_g is the refractive index of glass (unitless)

n_c is the refractive index of the coating (unitless)

θ_i is the incident angle in deg

θ_t is the transmitted angle in deg

$R_{coating}, R_{nocoating}$ is the reflected intensity with and without coating respectively (unitless)

$T_{coating}, T_{nocoating}$ is the transmitted intensity with and without coating respectively (unitless)

The incident angle is taken to be 45 degrees and the transmitted angle is 33 degrees. The refractive index of the coating is 1.3, the refractive index of the glass used to make the lens in the eyeglasses is 1.8 and the refractive index of air is 1. The chemical engineer finds that the transmitted intensity with the coating is 0.999 and that without the coating is 0.958.

Instructions

1. Rename the template file to match the format in the Deliverables list (see page 1).
2. Open the template file. Run the script to see how it works and what it displays.
3. Reformat the script to make it follow good programming standards.
4. Rerun the script to ensure your changes did not affect the final display of the two transmitted intensity values.
5. Submit your work in Gradescope:
    a. Open Gradescope > **A05 – All Problems**
    b. Find the file submission button for **Q2**.
    c. Submit **only** your m-file solution.
    d. When you have entered all the required information for the question, click the **Save Answer** button.

Problem 3: MATLAB Skills – Plots

Introduction

This problem allows you to practice plotting within MATLAB figures. You will submit images of your figures to Gradescope along with the script you used to create the figures.

Problem

For this problem, you will write a basic script that will plot data vectors in different figures. You will create three figures that are each formatted for technical presentation:

- Figure 1 will display a single data set on one set of axes
- Figure 2 will display two subplots, each with a different set of data on their own axes.
- Figure 3 will display two data sets together on one set of axes.

The problem generator **A05_skills.p** file will provide you with the data and units for your plots.

You need MATLAB open with both **A05_skills.p** and **A05Q3_plots_template.m** in your current folder to complete this problem.

Instructions

1. Open **A05Q3_plots_template.m** in the MATLAB Editor. You will use this template to create your script.
2. Type this command into the MATLAB Command Window prompt:
   ```
   >> A05_skills(PUID, 3)
   ```
   Where you replace PUID with your 8-digit Purdue University ID number (leave off the leading 00).
3. Review the data that displays to the Command Window. The remaining instructions are in this document.
4. Copy the variable assignments for `x`, `y1`, and `y2` to the `INITIALIZATION` of your script template and use the vectors to complete 3 figures:
   a. In the `FIGURE 1` section, write the code to plot y1 vs x into the first figure.
   b. In the `FIGURE 2` section, write the code to show two subplots vertically in the second figure.
      i. Plot y1 vs x in the top subplot.
      ii. Plot y2 vs x in the bottom subplot.
   c. In the `FIGURE 3` section, write the code to plot y1 vs x and y2 vs x on the same set of axes in the third figure.
   d. Format all plots and figures for technical presentation.
   e. Programming standards will not be graded in this script.
5. Run your script to generate the figure windows for each figure. Save each figure as an individual image file.
   a. In the figure window, click **File > Save As**.
   b. Select the location where you want to save the file. Choose a location that is easy for you find.
   c. Change the file format to Portable Network Graphics (PNG) and name the files using the format
      A05Q3_Figure1_*login*.png
      A05Q3_Figure2_*login*.png
      A05Q3_Figure3_*login*.png,
      where you replace *login* with your Purdue career account login name.
6. Submit your work in Gradescope:

a. Open Gradescope > **A05 – All Problems** and find the set of boxes that belong to **Q3**. Enter the required information along with your answer:
   - ☐ **Function call**. Copy the command that you entered at the command prompt to call the function and paste the full command into this box. Be sure your PUID is included.
   - ☐ **Instruction text**. Copy the instruction text that is displayed in the Command Window. Paste it into this box. Include all text provided.
   - ☐ **Solutions**. Submit the 4 requested files:
     1. The image file for Figure 1,
     2. The image file for Figure 2,
     3. The image file for Figure 3,
     4. And the script file with your code.
     5. **Make sure that each file has the correct name** (see page 1).

b. When you have entered all the required information for the question, click the **Save Answer** button.

Problem 4: Gate and Interconnect Delay

Introduction

Combine all the skills you have learned this week to create a script that follows good programming standards, imports data correctly, uses formatted plots to display information, and uses professional language to answer questions.

Problem

Context

Integrated circuits (IC), also known as chips, generally consist of many discrete circuit components such as transistors combined on a small, flat piece of semiconductor material. Engineers have worked to design integrated circuits that reduce the sizes of each component, which has allowed for dramatic size and weight reductions. To take advantage of the improvements, many chips now integrate various (hitherto discrete) components such as microprocessor, digital signal processors (DSPs), dedicated hardware processing engines, memories, and interfaces to I/O devices and off-chip storage in a single so called system on a chip (SoC). System-on-chip designs have the benefits of being energy efficient and compact, compared to more traditional hardware. For example, many wearable medical devices use system-on-chip technology so that patients can have small wearable devices with long battery lives. Figure 1 shows examples of several of these applications.

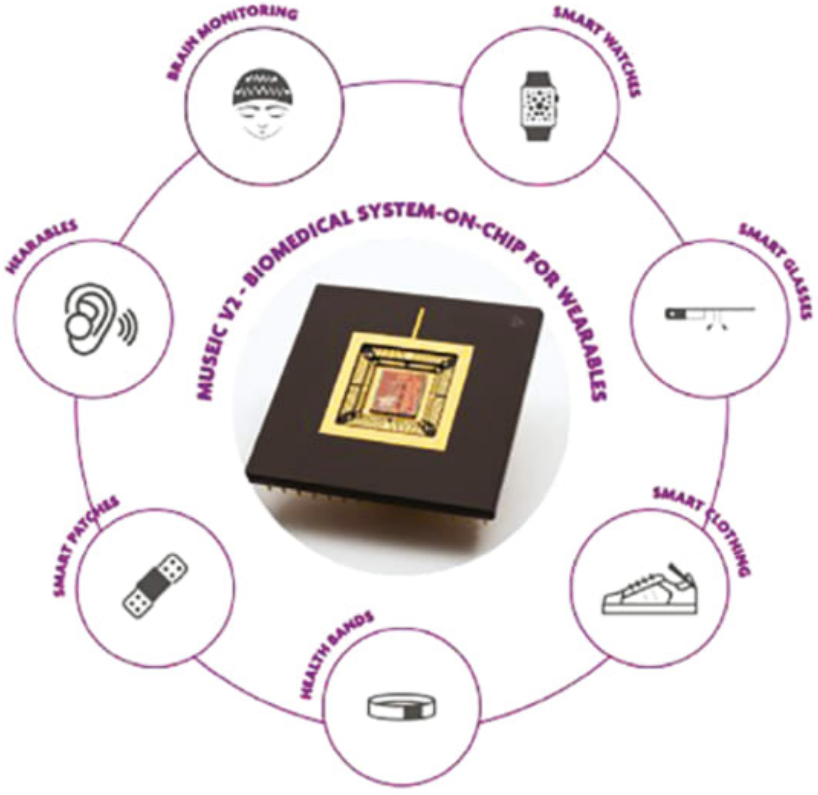


Figure 1. Examples of medical applications of system on chip designs

With increased functionality with each generation of new chips, the number of components, and therefore the number of components that signals need to travel through, grow. This increase causes delays. Interconnect delay, the delay caused from the travel time for information between distant components, can exceed the processing time of the chips themselves, known as the gate delay. Depending on the type of calculation required, the gate

delay, the interconnect delay, or their total may be the key factor limiting the speed of the processing of the system on chip. In this problem, you will analyze data comparing the gate delay and interconnect delay for several generations of integrated circuits. You will also be summing the delay for each type of integrated circuit to find the minimum overall delay.

Open **A05Q4_delays_template.m** in the MATLAB Editor. You will use this template to create your script. Import the data from **Delay_Data.csv** into your MATLAB script using an appropriate command and use the data to complete the following tasks.

1. Assign each column of data to a unique variable name
2. Create one figure that displays the following information on one set of axes:
    a. The gate delay data for each generation
    b. The interconnect delay data for each generation
3. Each data set can be modeled using the following equations. Add the model lines for each data set to the figure above, with each model as a smooth, continuous line:
    a. Interconnect delay: $y = 0.36e^{0.47x}$
    b. Gate delay: $y = 25.41e^{-0.34x}$

    Where, for both equations

    y = delay in picoseconds (ps)

    x = generation number
    c. **Note**: To create a smooth model line, you need to create an x vector that contains at least 50 elements. Otherwise, your line may not be smooth. Do not use the raw data to calculate the model delay values for this problem.
4. The total delay, which is the sum of the gate and interconnect delays, for each generation. Add the plot of total delay for each generation to the axes with the original data and models.
    a. Your plot should now have five (5) elements: raw data for interconnect and gate delays, model lines for interconnect and gate delays, and total delay data.
5. Find the minimum total delay value. **Use MATLAB code to determine the value.** Display the information to the Command Window using professional formatting.
    - Display decimal values with 3 decimal places.
    - Use a complete sentence.
6. Copy the text display for the minimum total delay and paste it as a comment in the `RESULTS` section of your MATLAB script.
7. Answer the following questions in the `ANALYSIS` section of your MATLAB script. Write your responses as MATLAB comments using the `%` at the front of each line:
    1. Which generation has the minimum gate delay? Which generation has the minimum interconnect delay? Justify your response.
    2. Which generation has the minimum total delay? Why is this different than either of the individual delays? Justify your response.
    3. What is an example of a situation when the original raw data would be most helpful? What is an example of a situation when the model would be most helpful?

Instructions

1. Rename the template file to match the format in the Deliverables list (see page 1).
2. Make sure your script follows good programming standards.

3. Run your completed script. It should generate one figure in a Figure Window and display the minimum total delay to the Command Window.
4. Save the figure as an image file.
    a. In the figure window, click **File > Save As**.
    b. Select the location where you want to save the file. Choose a location that is easy for you find.
    c. Change the file format to Portable Network Graphics (PNG) and name the files using the format

        A05Q4_Figure_*login*.png

        where you replace *login* with your Purdue career account login name.
5. Submit your work in Gradescope:
    a. Open Gradescope > **A05 – All Problems**
    b. Find the file submission buttons for **Q4**.
    c. Submit **only** your m-file, the image file of the figure, and the data file.
    d. When you have entered all the required information for the question, click the **Save Answer** button.

References:
Patti, R. S. (2006). Three-dimensional integrated circuits and the future of system-on-chip designs. Proceedings of the IEEE, 94(6), doi.10.1109/JPROC.2006.873612
https://www.medicaldesignbriefs.com/component/content/article/mdb/tech-briefs/26231

Confirm Your Submission

You should save your progress on each question so that you do not lose your progress. To confirm your answers, click the **Submit & View Submission** button at the bottom of the questions in Gradescope (or select the assignment name from the Gradescope dashboard, if you have already saved your answers and navigated away from the original submission page).

Confirm that your submission for **A05 – All Problems** includes

- ☐ The function call and instruction text for each skills question (Q1 and Q3);
- ☐ The expected deliverables and results for all questions;
- ☐ Correct file names for any submitted files, including your Career Account login at the end where required.

Assignment Grading

Your work will be graded using the evidences given in the course learning objectives. Familiarize yourself with the LOs and their evidences listed for each problem, which are below. Each problem's assignment grading has a table and a flowchart. The table outlines what LOs will be used to grade your work and what point values are assigned to each evidence item. The flowchart outlines the grading process that a grader will use to assess your work.

Find the list of the course LOs, with evidences, on Brightspace (Content > Key Course Info > Learning Objectives).

Problem 1

LOs: PC05, MAT03

Problem 1 is worth 5 points, where each part is 1 point. There is some partial credit on parts B-E. The partial credit may be more specific than what is in the course LOs and is based on evidence MAT03 (1) and (5).

You must meet the PC05 expectations for each question. If you do not meet these, you will lose additional credit.

Evidence	Penalty
PC05 (1)	Lose full credit on problem
PC05 (2)	Lose 25% of full credit on problem
PC05 (3)	Lose 25% of full credit on problem
PC05 (8)	Lose 10% of full credit on problem

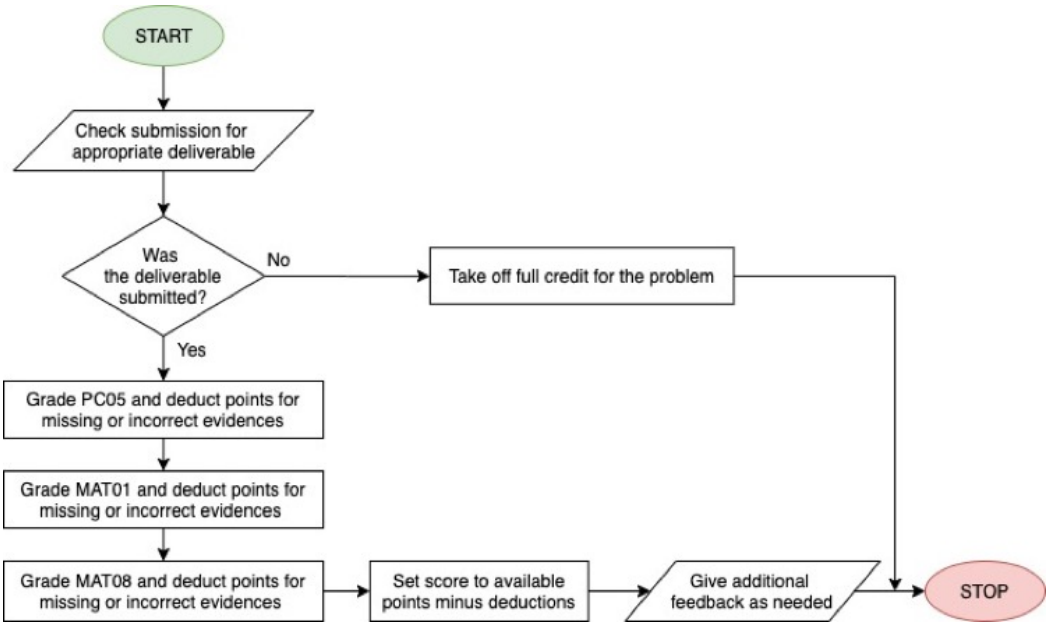
Problem 2

LO Table:

Note: PC05 evidences are only deductions since you are expected to follow the assignment instructions.

Evidence	PC05	MAT01	MAT08
(1)	-5	0.4	0.0
(2)	0	0.0	0.2
(3)	-1.25	0.4	0.2
(4)	-0.25	0.4	0
(5)	0	0.4	0
(6)	0	0.4	0
(7)	0	0.3	0
(8)	0	0.3	0

Grading Process:



Problem 3

LOs: PC05, MAT03, EPS02

Problem 1 is worth 5 points, where Figure 1 is worth 1.5 points, Figure 2 is worth 1.25 points, and Figure 3 is worth 1.25 points. Partial credit may be more specific than what is in the course.

You must meet the PC05 expectations for each question. If you do not meet these, you will lose additional credit.

Evidence	Penalty
PC05 (1)	Lose full credit on problem
PC05 (2)	Lose 25% of full credit on problem
PC05 (3)	Lose 25% of full credit on problem
PC05 (4)	Lose 5% of full credit on problem

Problem 4

You must submit the m-file with plotting code to receive any credit on this problem.

LO Table:

Note: PC05 evidences are only deductions since you are expected to follow the assignment instructions.

	PC05	MAT01	EPS02	EPS01	MAT08	MAT02	MAT03
(1)	-5	0.1	0.2	0.1	0.2	0.2	0.3
(2)	-0.5	0.1	0.2	0.2	0.4	0.2	0.1
(3)	-1.25	0.1	0.2	0.1	0.2	0.2	0.1
(4)	-0.25	0.1	0.2	0.1	0	0	0.1
(5)	0	0.1	0.2	0.1	0	0	0
(6)	0	0.1	0.2	0	0	0	0
(7)	0	0.1	0.2	0	0	0	0
(8)	0	0.3	0	0	0	0	0

Grading Process: