# A06 · Plots & Logic Operations

## Introduction

### Assignment Goals

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. You will learn to code logical statements with MATLAB syntax and then you will apply those skills to an engineering context.

### Successful Completion

This assignment has four (**4**) problems. Problems 1 and 3 go with Class A; Problems 2 and 4 go with Class B.

1. Read *Notes Before You Start*, on **Page 1**.
2. Read each problem carefully. You are responsible for following all instructions within each problem.
    a. The deliverables list within each problem contains everything you are expected to submit.
    b. You need the problem generator **A06_skills.p** for Problems 1 and 2. See this link to view instructions.
3. Complete the problems using the problem-specific m-file templates when a template is provided in the assignment download.
4. For any file, replace *template* or *login* in the filename with your Purdue Career Account login.

    ***Example***: `A06Prob1_context_template.m` will be renamed `A06Prob1_context_pboilerm.m` for a student whose Career Account login is pboilerm.
5. Review your work using the learning objective evidences.
6. When your work is complete, confirm your deliverables are submitted to Gradescope.
    a. Note the two different assignments in Gradescope. See *Notes Before You Start*.
        i. Submit Problems 1 and 2 to **A06 – Skills Problems**.
        ii. Submit Problems 3 and 4 to **A06 – Context Problems**.
    b. You can resubmit to Gradescope as many times as you need. Only your final submission is graded.
    c. Do **NOT** upload any document not listed in the deliverables. Do not upload temporary versions of m-files (*.m~ or *.asv) – these files will be ignored by Gradescope.
7. Late submissions will be accepted up to 24 hours after the due date and will result in a 25% penalty.

### Learning Objectives & Grading

This course uses learning objectives (LOs) to assess your work. You can find a full list of the course LOs here. Review the grading outline at the end of each problem in this assignment to see each problem's LOs.

## Notes Before You Start

### Helpful MATLAB Commands

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

```
sgtitle, plot, title, xlabel, ylabel, legend, close, linspace,
min, max, mean, find, any, all
```

## 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.

## 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.
    - MATLAB plots discrete points on a graph: a line is created by connecting the points. You want a smooth line when you plot a model, so you must have enough points in the independent variable vector when you calculate the dependent variable vector. Having enough points allows curved lines to display smoothly. Note that the number of points necessary to make a smooth model line can vary, depending on the data set. A good rule of thumb is to create with an x-vector that starts with the lowest x data value, ends with the largest x data value, and has 50 points.
- 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.

## `find` Command in the MATLAB Editor

If you use the `find` command within the MATLAB editor, you may notice that MATLAB sometimes produces a warning on the lines with the `find` command. MATLAB may suggest that you use logical indexing, which allows you to use logical 1s and 0s to identify which values in a vector correspond to the ‘true’ condition, similar to array indexing. You can use either method, `find` or logical indexing, when appropriate. There may be times where only one of them works. You should know how the `find` command works for exams.

## Intervals and Ranges

A range of numbers can be defined as an interval. If a boundary value is included within the range, then we say the range is “inclusive” of the number. If a boundary value is not included in the range, then the range is “exclusive” (or not inclusive) of that number. See examples below for the range of number 5 to 10.

| Relational notation | Interval notation | Explanation |
|---|---|---|
| $5 \leq x \leq 10$ | $[5, 10]$ | The numbers between 5 and 10, inclusive of both 5 and 10 |
| $5 \leq x < 10$ | $[5, 10)$ | The numbers between 5 and 10, inclusive of only 5 (exclusive of 10) |
| $5 < x \leq 10$ | $(5, 10]$ | The numbers between 5 and 10, inclusive of only 10 (exclusive of 5) |
| $5 < x < 10$ | $(5, 10)$ | The numbers between 5 and 10, exclusive of both 5 and 10 |

## Gradescope

You will submit all your deliverables to Gradescope for grading.

- This assignment has two submission locations in Gradescope: one for skills problems and one for context problems. Read the assignment deliverables carefully to see where to submit.
- Skills problems are “Online assignments” in Gradescope. View the Gradescope help for online assignments if you need assistance with submitting your work.
- Context problems are “Programming assignments” in Gradescope. You must submit ALL files for the assignment at once into the assignment (i.e., all files for all problems in the assignment). This is true for your initial submission and for any resubmitted work. Click here for help.
- If you cannot copy-paste into Gradescope, then you may need to refresh or update your browser. Use Google to find a solution for your specific browser.

### 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 your section’s Gradescope course.
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. You can access Gradescope through Brightspace throughout the semester.

# Problem 1: 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 Generator Information

If you have questions about how to use the problem generator, review this link that shows step-by-step instructions.

| File Name | PUID | Problem Number |
|---|---|---|
| A06_skills | Your 8-digit PUID | 1 |

### Submission

| | | | |
|---|---|---|---|
| Gradescope Assignment | A06 – Skills Problems | Assignment Type | Individual |
| Deliverables | □ A06Q1_plots_*login*.m<br>Supporting files:<br>□ A06Q1_figure1_*login*.png<br>□ A06Q1_figure2_*login*.png<br>□ A06Q1_figure3_*login*.png | | |

## 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 file will provide you with the data and units for your plots. The remaining instructions are in this document.

1. Enter the problem generator function call into the MATLAB Command Window prompt.
2. Review the data that displays to the Command Window.
3. Open **A06Q1_plots_template.m** in the MATLAB Editor. You will use this template to create your script.
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. For Figure 2, add a title to the subplot grid using the `sgtitle` command and use short, concise titles for the individual subplots.

   e. Programming standards will not be graded in this script but do organize your work in the provided sections within the template.

5. Run your script to generate the figure windows for each figure. Save each figure as an individual image PNG file and use an appropriate file name. See this video for help.
6. Submit instruction text with the run receipt and the required file deliverables to Gradescope. See this video for help (this is an untimed assignment). Do not submit any additional files.
   a. Make sure your files have the correct names and format. See the deliverables list.

## Grading

LOs: PC05, MAT03, EPS02

Point value: 7 points. The partial credit may be more specific than what is in the course LOs and is based on evidence EPS02. 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 (file name) |
| PC05 (4) | Lose 10% of full credit (file type) |
| PC05 (8) | Lose 10% of full credit |

# Problem 2: MATLAB Skills – Logic Statements

## Introduction

This problem allows you to practice writing logical statements using built-in functions and relational and logical operations on a matrix.

### Problem Generator Information

If you have questions about how to use the problem generator, review this link that shows step-by-step instructions.

| File Name | PUID | Problem Number |
|---|---|---|
| A06_skills | Your 8-digit PUID | 2 |

### Submission

| | | | |
|---|---|---|---|
| Gradescope Assignment | A06 – Skills Problems | Assignment Type | Individual |
| Deliverables | □ Requested information and solutions | | |

## Problem

This problem has five (5) parts.

1. Enter the problem generator function call into the MATLAB Command Window prompt. This will generate the instructions for each part of this problem.
    a. This generator will also assign two variables to your Workspace. The variables are named mat1 and mat2. Use these variable names to complete this problem.
2. Read the written instruction text that appears in the Command Window. Use MATLAB to check your answers.
    a. Do not use any looping or selection structures. You can and should complete each question using logic statements and built-in functions.
        i. Each question can be answered with a single line of code, but you are allowed to use more than one command if they are in the proper order.
    b. You may hardcode any values given in the question text but do not hardcode values of `mat1` or `mat2` into your commands.
    c. Assign all calculations to variables. Make sure the variables are valid in MATLAB, but you are not required to use detailed variable names as described in the programming standards. Variable names like `partA` are acceptable.
    d. Your commands must run in MATLAB to receive full credit.
    e. Programming standards will not be graded. Do not include comments in your solutions.
    f. Only submit the commands to Gradescope, not any numeric results.
3. Submit your instruction text with run receipt and solutions into Gradescope. Follow any extra instructions that appear in Gradescope.
4. Save your answers in Gradescope. See this video for help (this is an untimed assignment).

## Grading

LOs: PC05, MAT02, MAT03

Point value: 7 points. The partial credit may be more specific than what is in the course LOs and is based on evidence MAT02(1), MAT03 (1), (2), (3), 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 3: Gate and Interconnect Delay (Plots)

## Introduction

Combine all your plotting and script skills 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.

### Submission

| | | | |
|---|---|---|---|
| Gradescope Assignment | A06 – Context Problems | Assignment Type | Individual |
| Deliverables | □ A06Q3_delays_*login*.m<br>Supporting files:<br>□ A06Q3_figure_*login*.png<br>□ Delay_Data.csv | | |

## 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.

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.

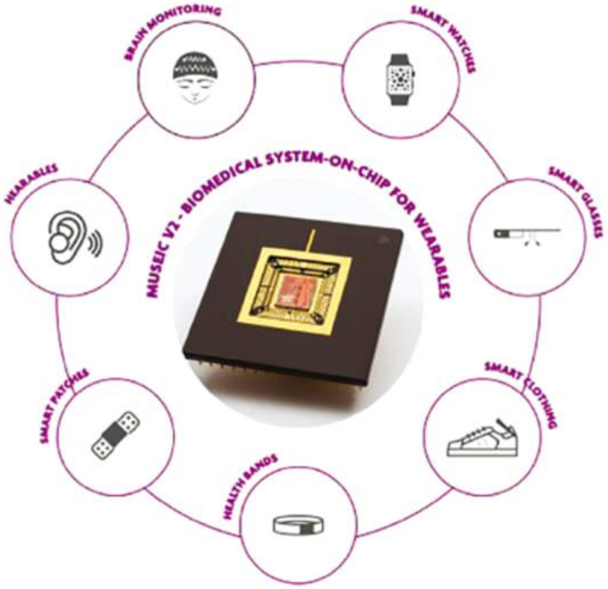


*Figure 1. Examples of medical applications of system on chip designs*

**Instructions**

Open **A06Q3_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**: See the information about smooth model lines in *Notes before you start*. 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 1 decimal places.
    - Use a complete sentence.
6. Run your script once you have successfully debugged it and have it working how you want. 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?

**Submit your work**

1. Rename the template file to match the format in the Deliverables list.
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. See this video for help.
5. Submit your m-file, final image, and data file to Gradescope once all your context problems are complete. Click here for help.

## 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

## Grading

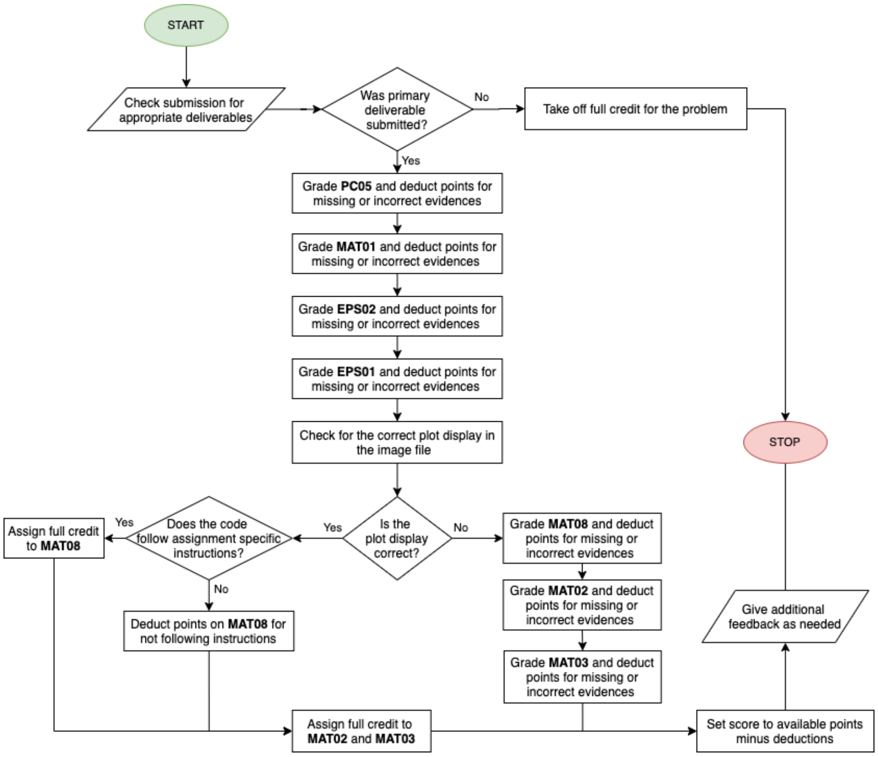
LOs: PC05, MAT01, ESP02, EPS01, MAT08, MAT02, MAT03

Point value: 8 points. Partial credit is possible; see the LO table below for details. You must meet the PC05 expectations. If you do not meet these, you will lose additional credit.

### LO Table

| | PC05 | MAT01 | EPS02 | EPS01 | MAT08 | MAT02 | MAT03 |
|---|---|---|---|---|---|---|---|
| (1) | -100% | 0.2 | 0.3 | 0.2 | 0.4 | 0.2 | 0.3 |
| (2) | -25% | 0.1 | 0.3 | 0.4 | 0.8 | 0.2 | 0.1 |
| (3) | -10% | 0.1 | 0.5 | 0.2 | 0.4 | 0.2 | 0.1 |
| (4) | -15% | 0.3 | 0.3 | 0.2 | 0 | 0 | 0.1 |
| (5) | 0 | 0.1 | 0.3 | 0.2 | 0 | 0 | 0 |
| (6) | 0 | 0.3 | 0.5 | 0 | 0 | 0 | 0 |
| (7) | 0 | 0.1 | 0.2 | 0 | 0 | 0 | 0 |
| (8) | 0 | 0.4 | 0 | 0 | 0 | 0 | 0 |

### Grading Process:


START
Check submission for appropriate deliverables
Was primary deliverable submitted?
No
Take off full credit for the problem
Yes
Grade **PC05** and deduct points for missing or incorrect evidences
Grade **MAT01** and deduct points for missing or incorrect evidences
Grade **EPS02** and deduct points for missing or incorrect evidences
Grade **EPS01** and deduct points for missing or incorrect evidences
Check for the correct plot display in the image file
STOP
Assign full credit to **MAT08**
Yes
Does the code follow assignment specific instructions?
Yes
Is the plot display correct?
No
Grade **MAT08** and deduct points for missing or incorrect evidences
No
Deduct points on **MAT08** for not following instructions
Grade **MAT02** and deduct points for missing or incorrect evidences
Give additional feedback as needed
Grade **MAT03** and deduct points for missing or incorrect evidences
Assign full credit to **MAT02** and **MAT03**
Set score to available points minus deductions

## Problem 4: Landslide Monitoring (Logic)

### Introduction

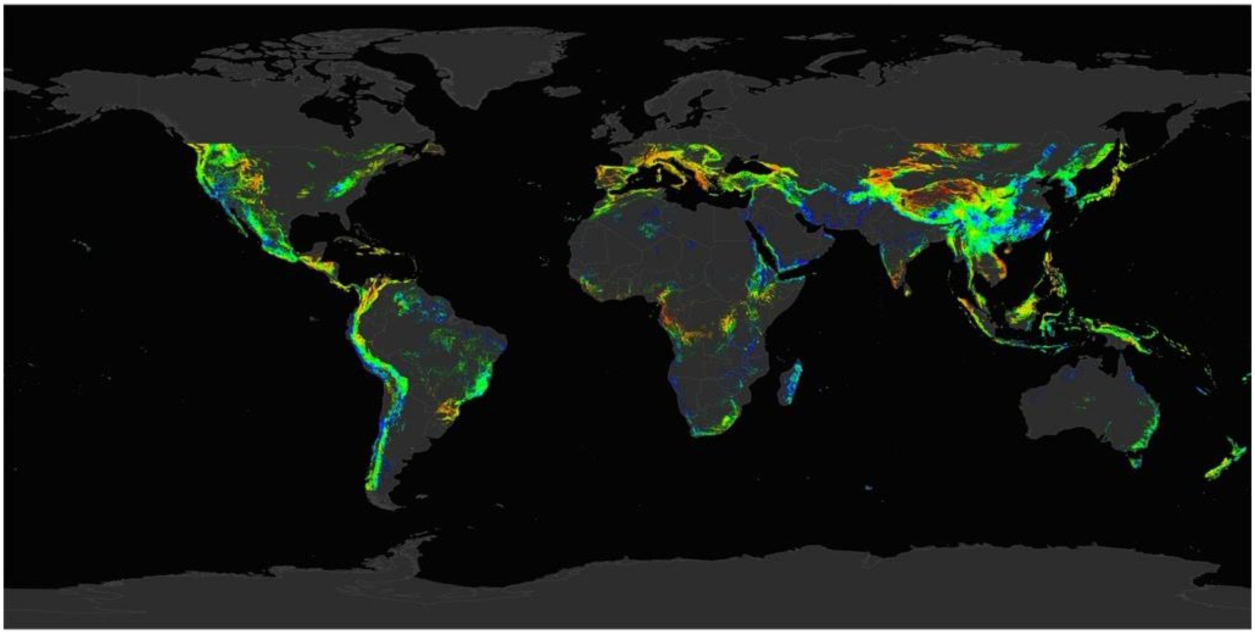
This problem focuses on your ability to write a MATLAB script that uses relational and logical operations to answer engineering-based questions. You will also use programming commands to display text and numeric results to the Command Window in a professional manner. Be sure to follow good programming standards in your script.

### Submission

*Individual*

| | | | |
|---|---|---|---|
| Gradescope Assignment | A06 – Context Problems | Assignment Type | Individual |
| Deliverables | ☐ A06Q4_landslide_*login*.m<br>☐ Supporting files: Data_global_landslide.csv | | |

### Problem



A civil engineer has been tasked with sorting and understanding landslide data from around the world. You will practice using relational and logical operators with the data file named **Data_global_landslide.csv** to help the civil engineer answer some important questions about the data.

The data file contains 8 columns of data. There are three categories of landslides in the columns labelled "landslide category", each with a category number. The same is true for the columns "landslide trigger", which contains numbers representing what triggered each landslide, and "landslide size" which tells us whether the landslide was small, medium or large. The table below explains what the different numbers mean for each column:

| Landslide category | Landslide trigger | Landslide size |
|---|---|---|
| 1: landslide | 10: rain | 100: small |
| 2: mudslide | 20: continuous rain | 200: medium |
| 3: snow avalanche | 30: downpour | 300: large |
| | 40: snow fall or snow melt | |

Import the data file into MATLAB using commands within your script and use it to answer the following questions.

1. How many landslides occurred that were categorized as “mudslide” whose landslide trigger was not “downpour”, or landslides that were triggered by “downpour” but were not categorized as “mudslide”?
2. What was the average latitude of the landslides that were categorized as “snow avalanche”?
3. How many cases involved any type of landslide with fatalities greater than 50 that occurred within latitudes greater than 0 or longitudes less than 0 but not both?
4. How many “mudslides” and “landslides” occurred due to “snow fall or snow melt” that had landslide sizes that were at least “medium” with latitudes above the equator? (The equator has a latitude of 0.)
5. How many and in which year(s) were there “medium” sized landslide categories of “snow avalanche” in the USA? You know that the latitude for USA should range between approximately 19.5 to 64.86 while the longitude should be negative.

Display your results using professional and easily read format. *Display Guidelines:*

- Print values within an appropriate sentence that answers the question.
- Use appropriate numerical formatting (e.g., control the number of decimal places displayed.)
- Identify the question number in the text display.

**NOTE**: Do not use any loops or other non-sequential structures to answer these questions.

Once your code is working, run it to display the text results. In the `RESULTS` section of the template, paste your text display for each question as comments.

### Instructions

1. Read through the entire problem statement.
2. Complete your script, run it to get your results, and paste those results as comments into the script.
3. Submit your properly named m-file and data file to the appropriate Gradescope assignment (see the submission list at the beginning of this problem).
    a. Do not submit any other files.
    b. You need to submit all files for all problems in this assignment. Click here for help
4. **Important Note**: If you need to resubmit your work to Gradescope, you must submit all files (even if only one file changed). For example, this problem requires a data file and an m-file. If you submit both, then make changes to your m-file and resubmit. Click here for help.

## References

https://svs.gsfc.nasa.gov/4710

## Grading

LOs: PC05, MAT01, EPS01, MAT08, MAT02, MAT03

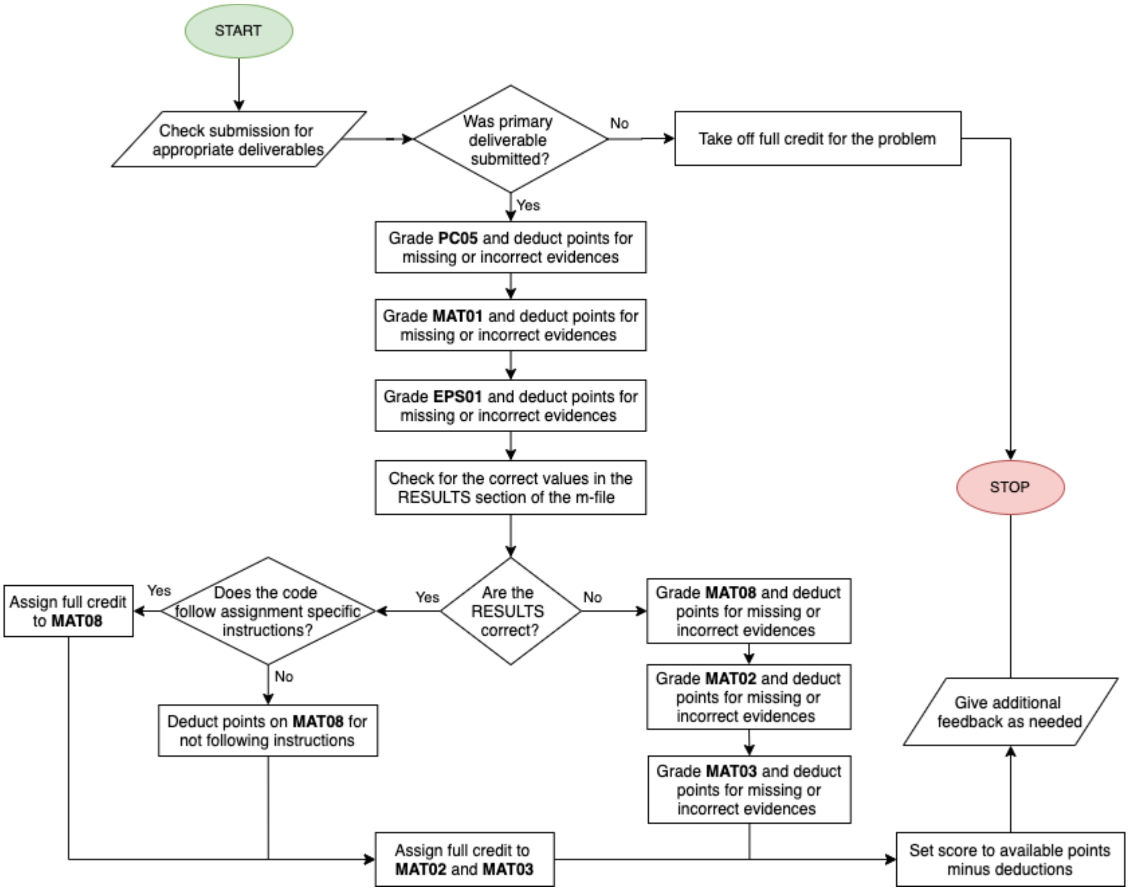
Individual assignment point value: 8 points. Partial credit is possible; see the LO table below for details. You must meet the PC05 expectations. If you do not meet these, you will lose additional credit.

### LO Table

| | PC05 | MAT01 | EPS01 | MAT08 | MAT02 | MAT03 |
|---|---|---|---|---|---|---|
| (1) | -100% | 0.4 | 0 | 0.8 | 0.2 | 0.2 |
| (2) | -25% | 0.2 | 0 | 0.8 | 0.2 | 0.2 |
| (3) | -10% | 0.2 | 0.4 | 0.4 | 0.2 | 0.2 |
| (4) | -5% | 0.4 | 0.6 | 0 | 0 | 0.2 |
| (5) | 0 | 0.2 | 0.4 | 0 | 0 | 0.8 |

| | | | | | | |
|---|---|---|---|---|---|---|
| (6) | 0 | 0.2 | 0 | 0 | 0 | 0 |
| (7) | 0 | 0.2 | 0 | 0 | 0 | 0 |
| (8) | 0 | 0.6 | 0 | 0 | 0 | 0 |

### Grading Process


START
Check submission for appropriate deliverables
Was primary deliverable submitted?
No
Take off full credit for the problem
Yes
Grade PC05 and deduct points for missing or incorrect evidences
Grade MAT01 and deduct points for missing or incorrect evidences
Grade EPS01 and deduct points for missing or incorrect evidences
Check for the correct values in the RESULTS section of the m-file
STOP
Are the RESULTS correct?
Yes
No
Does the code follow assignment specific instructions?
Yes
Assign full credit to MAT08
No
Deduct points on MAT08 for not following instructions
Grade MAT08 and deduct points for missing or incorrect evidences
Grade MAT02 and deduct points for missing or incorrect evidences
Grade MAT03 and deduct points for missing or incorrect evidences
Give additional feedback as needed
Assign full credit to MAT02 and MAT03
Set score to available points minus deductions