General Instructions

Due Date: Sunday, October 16th, 2022 by 11:59pm (submit via Zybooks)

Assignment Instructions:
This assignment has one problem, summarized below. You will use MATLAB as a tool to solve the problem for the given test cases, ensuring that your code is flexible for any additional test cases that might be used to evaluate it.

- Beam Deflection with Uniform Load Distribution (Application to Civil and Mechanical Engineering)

zyBooks Submission Instructions:
After completing this assignment in MATLAB, to receive credit, you must submit your code in Zybooks. The following components must be submitted under the specified chapter of the course Zybooks before the deadline to receive credit.

- Chapter 34.1 MA5: ValidInput Function
- Chapter 34.2 MA5: Main Script

To submit your script, copy and paste your code into the submission window, making sure to remove any housekeeping commands. You may submit to Zybooks as many times as you want before the deadline, without any penalty. The highest score attained before the deadline will be graded. All components are due before the due date. No credit will be given if it is not submitted through the Zybooks platform before the deadline. Credit for each component will be awarded based upon the percentage of successfully completed assessments.

Explanation of P-Code:
Under the Additional Resources folder accompanying this prompt, you will find a file named **ValidInput_Solution.p**. This is a working solution to the ValidInput function required to complete this problem. This file is unopenable and the contents can't be read in a text editor, but it can be called like a typical .m function file in MATLAB. If you get stuck and are unable to successfully develop your own Brightness function, you are encouraged to instead copy ValidInput_Solution.p to your working folder and rename it **ValidInput.p**. Now, when your main script calls the ValidInput function, it will automatically call the .p file, and you will now have an opportunity to successfully finish developing your main script.

Proficiency Time: Times are included with the Background and Task sections. These times are the estimated amount of time it should take you to **redo** an assignment once you are fully proficient in material that it covers. To practice, reread the background in the given Comprehension Time and attempt to complete the problem in the given Proficiency Time.

Academic Honesty Reminder

The work you submit for this assignment should be your work alone. You are encouraged to support one another through collaboration in brainstorming approaches to the problem and troubleshooting. In this capacity, you are permitted to view other students' solutions, however, copying of another student's work is strongly discouraged.

This assignment will be checked for similarity using a MATLAB code. The similarity code will check each submission for likeness between other student submissions, past student submissions, the solution manual, and online resources and postings. If your submission is flagged for an unreasonably high level of similarity, it will be reviewed by the ENGI 1331 faculty, and action will be taken by faculty if deemed appropriate.

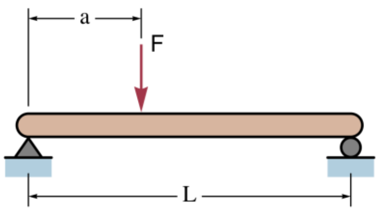
NOTE: Since this is an automated system for all sections, if any of your work is not your own, you will be caught. Changing variable names, adding comments, or spacing will not trick the similarity algorithm.

Background

Comprehension Time: 10 – 15 min

This problem will introduce you to an example of a simply supported beam subject to a load modeled as a point force. The background provides all of the equations and variable definitions needed to achieve the tasks described.

Below represents a simply supported beam of length L subject to a force F.



The beam has a width w and height h. The beam also has a Young's modulus, E, which represents the ability of a material to withstand changes in length (sometimes referred to as modulus of elasticity). A data file (**MaterialElasticity.mat**) is provided that contains materials with associated Young's moduli.

For a single force the deflection of the beam is determined using the derived equation

$$y(x) = \begin{cases} -\theta x + \frac{Rx^3}{6EI}, & x \le a \\ -\theta x + \frac{Rx^3}{6EI} - \frac{F}{6EI}(x-a)^3, & x > a \end{cases} \qquad \text{Eq. 1}$$

where I is the area moment of inertia of the cross section, R is the reaction force on the beam at the left end, and θ is the clockwise rotational angle of the beam at the left end. These have already been derived for you and are given by the following equations:

$$I = \frac{wh^3}{12} \qquad \text{Eq. 2}$$

$$R = \frac{F}{L}(L-a) \qquad \text{Eq. 3}$$

$$\theta = \frac{Fa}{6EIL}(2L-a)(L-a) \qquad \text{Eq. 4}$$

Following the tasks below, you will develop a program that will calculate the deflection of this simply supported beam with one or more loads for any beam material, dimensions and loading. You may assume that the given values will always be in SI units; this implies that the deflection equation above will calculate the deflection in meters. See **MA5_SampleCalculations.pdf** for examples of these beam deflection calculations.

Tasks Proficiency Time: 50 – 70 min

TASK 1: Material selection and data validation (3 – 5 min)
Prompt the user to select a material from the **Material** string array. If user exits out of the menu, keep asking the user for a selection until one is made. You will use this selection to determine the Young's modulus value from the **Elasticity** vector. Each material has a corresponding Young's modulus [Pa].

TASK 2: Dimensional information and data validation (13 – 17 min)
Write a user-defined function named **ValidInput.m** which validates a user's input. This function should use the same syntax as the built-in input() function with one input (the prompt) and one output (the stored variable). In the function, perform the following data validation: if the entered value is zero or negative, prompt the user for a new value. If after 3 attempts the user has not entered a positive value, produce a warning and take the absolute value of the most recently entered value. If the most recent value is zero, produce an error and terminate the program.

Call this function 3 times in your main script to prompt the user for the length L [m], the width w [m], and the height h [m] of the beam.

TASK 3: Calculate deflection for a single load on a simply supported beam (18 – 24 min)
Prompt the user to enter a concentrated force F [N] and the location of the force on the beam a [m]. Include the length of the beam L in the statement requesting the location of the force. Using Eq. 1 through 4, determine the deflection curve across the entire beam and plot the deflection as a solid line. Your plot must include the following:

- X-axis label
- Y-axis label
- X-axis limit set to the length of the beam
- Deflection shown in **millimeters**
- Title with material name
- Gridlines

HINT: See **MA5_SampleCalculations.pdf** for an example of using Eq. 1 through 4 to determine the deflection at a single point on the beam. This calculation can be generalized to apply to the entire beam.

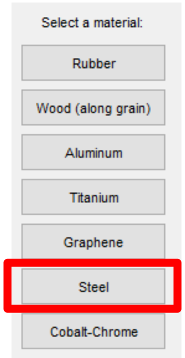
TASK 4: Calculate deflection for multiple loads on a simply supported beam (13 – 19 min)
Prompt the user to enter multiple concentrated forces [N/m] and the locations of each of the forces on the beam [m] as vectors. Using Eq. 1 through Eq. 4, determine the total deflection of the beam caused by all of the forces across the entire beam and plot the deflection as a solid line in a new figure. Your plot should include the same formatting as the plot in Task 3. Then, output to the command window the maximum deflection of the beam in millimeters.

HINT: The total deflection of the beam at any point is the sum of the deflections caused by each single force at that point.

TASK 5: Programming controls (3 – 5 min)
After Task 4, ask the user if they would like to repeat the multiple loads calculation with different forces and locations. If yes, repeat Task 4. If not, your program should end.

Sample Output

Case 1: Sample of warnings and errors from Task 2

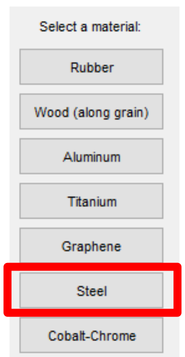


Command Window

```
Enter the length of the beam [m]: -8
Enter the length of the beam [m]: -8
Enter the length of the beam [m]: -8
Warning: Invalid value entered 3 times. Taking the absolute value of the entered value.
> In ValidInput (line 10)
In MA6_cougarnetID (line 16)
Enter the width of the beam [m]: 0
Enter the width of the beam [m]: -5
Enter the width of the beam [m]: 0
Error using ValidInput (line 13)
Entered value is zero. Cannot solve problem, exiting program.

Error in MA6_cougarnetID (line 17)
w = ValidInput('Enter the width of the beam [m]: ');
```

Case 2: Sample of fully ran program, with Task 4 repeated once



Command Window

```
Enter the length of the beam [m]: 10
Enter the width of the beam [m]: 0.1
Enter the height of the beam [m]: 0.1
Enter the magnitude of a concentrated force acting on the beam [N]: 1500
Enter the location of the force (0 - 10.00 meters): 7

Enter the magnitudes of multiple loads on the beam [N]: [800 350 200]
Enter the locations of the forces, in order (0 - 10.00 meters): [3 5 8]
The maximum deflection of the beam is 13.748 [mm].

Enter the magnitudes of multiple loads on the beam [N]: [800 350 200 1500]
Enter the locations of the forces, in order (0 - 10.00 meters): [3 5 8 7]
The maximum deflection of the beam is 28.597 [mm].
```

First figure displayed

Second figure displayed

Second figure re-displayed