MEMORANDUM

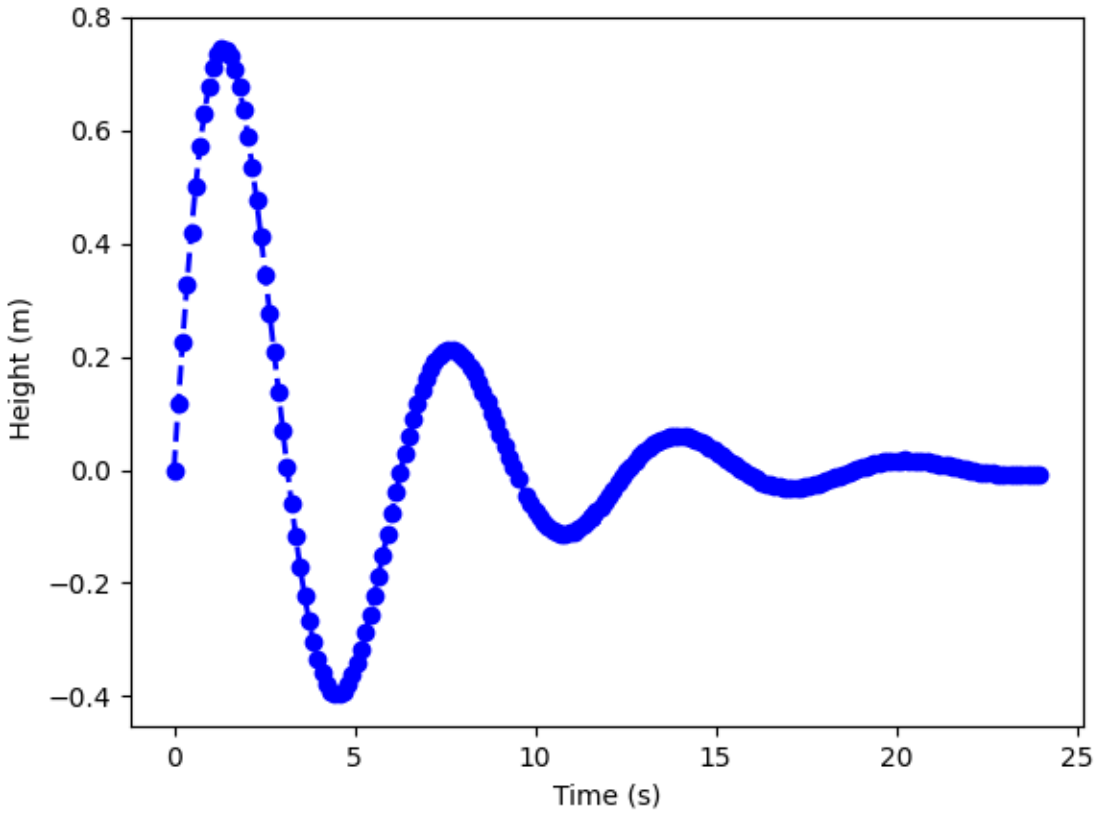
To: Charlie Refvem, Lecturer, Department of Mechanical Engineering, Cal Poly SLO
crefvem@calpoly.edu

From: Michael Shokoohi
msshokoo@calpoly.edu

Date: 09/28/2025

RE: **ME-405-01 mecha 02**

In this assignment I created a python script that is able to read in a .csv file, flag lines that have comments, lacked 2 data points, or had characters/symbols. Below is the plot output from the script as well as the flagged lines printed to the console.



```
There are less than 2 elements on row 6
There is a comment on row: 7
There are less than 2 elements on row 8
There are less than 2 elements on row 12
There is a comment on row: 86
```

Below is the python script written for this assignment:

```
'''

Author: Michael Shokoohi

Term: Fall 2025

Course: Mechatronics ME 405

Assignment Description: Write a python Script that is capable of reading a
csv file and plotting the data

                        Note that the csv file should be able to have cells
with white space, comments, or text

                        that are ignored by script but flags the row with
error type.
'''

from matplotlib import pyplot as plt


#Initializing lists to hold X data from column 1 and Y data from column 2

X_Data=[]

Y_Data=[]


def ReadCsv(File):
  '''

  Input: a csv file that needs to be read in format r"The file directory you
want in windows format"


  Description: Only data in the first two columns of each row are converted
to type float and stored in lists to be output. Note that the

  first row is presumed to be the header and is therefore only used for
labeling the x and y axis so the values in those cells are neglected.
```

```
  Output: True on successful execution, X_Data (list of float), Y_Data (list
of float), X-axis label (string), Y-axis label (string)
  '''


  # Opening the csv Data file
  with open(File,'r') as f:
    # Grabbing the header of the file
    header= f.readline().strip().split(',')

    # Iterating through all rows of the csv file
    for id, row in enumerate(f):
        RawData=row.strip().split(',')

        # Checking if the vales in the first 2 columns of each row
        # are able to be converted to float.
        try:
          float(RawData[0])
          float(RawData[1])

        except ValueError:
          #Detecting if there is a comment in the line
          if "#" in row:
             print(f"There is a comment on row: {id+2} ")
             continue

          # Detecting if there are atleast 2 elements in the row
          if len(RawData)<2 :
            print(f"There are less than 2 elements on row {id+2}")
            continue

          # Detecting if there are any letters or symbols present that are
not part of a comment
          if not RawData[0].isdigit() or not RawData[1].isdigit():
```

```
            print(f"There is either a letter of symbol on row: {id+2}. And
it is not part of a comment")
            continue


        # Creating the lists for X and Y data
        X_Data.append(float(RawData[0]))
        Y_Data.append(float(RawData[1]))


    return True, X_Data, Y_Data, header[0], header[1]


def PlotData(X_Data: list,Y_Data: list,X_Label:str,Y_Label:str):
    """
    Input: X Y data as lists and the X Y axis labels as strings


    Description: Creates and displays a plot of Y vs. X
    """
    # Creating plot
    plt.plot(X_Data,Y_Data,color='Blue', linewidth=2, linestyle='--',
marker='o')
    plt.xlabel(X_Label)
    plt.ylabel(Y_Label)


    # Displaying graph
    plt.show()




if __name__== "__main__":
  #Defining the csv file to read (File is within the same directory)
  File= 'data.csv'


  #Reading and collecting data from csv
  _, X_Data, Y_Data,X_Label,Y_Label=ReadCsv(File)
```

```
#Plotting the collected data
PlotData(X_Data,Y_Data,X_Label,Y_Label)
```