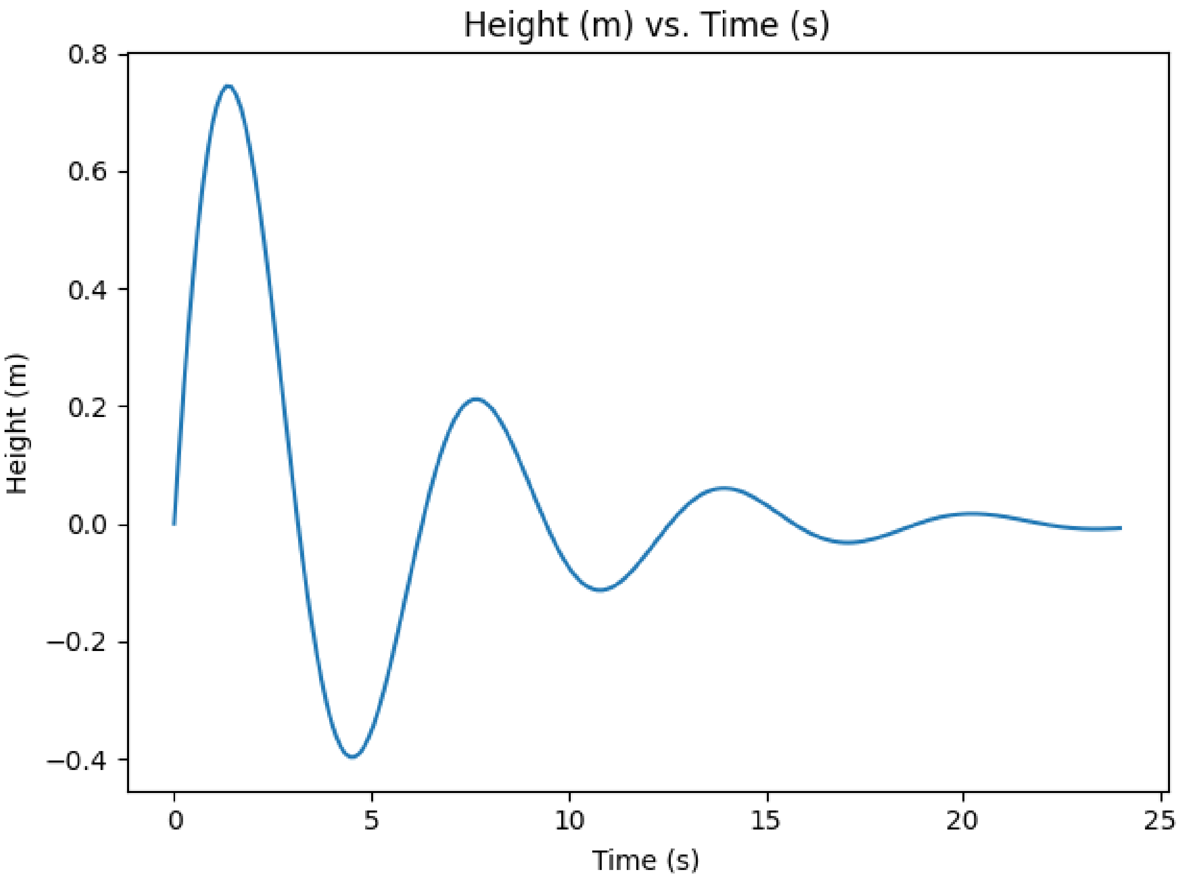
Mechanical Engineering
CAL POLY

To: Dr. Charlie Refvem, Mechanical Engineering, Cal Poly SLO, crevem@calpoly.edu
From: Caiden Bonney, clbonney@calpoly.edu
Date: September 22, 2025
RE: Homework 0x00 - Reading CSV Files and Using matplotlib

Produced Plot



Log of Rejected Rows

```
Line 6 rejected due to empty line:
Line 7 rejected due to full-line comment: # 0.0 0.0 0.0, 0.0
Line 8 rejected due to empty line:
Line 12 rejected due to not enough entries: waffles!!!
```

Code

```
from matplotlib import pyplot as plt

# Function definition
def reject(reason, linenum, line):
    print(f"Line {linenum} rejected due to {reason}: {line}")

# Variable initialization
file_name = "data.csv"
col_number = []
column1 = []
column2 = []
first = True

try:
    # Open the file in read mode
    with open(file_name, 'r') as f:
        content = [l.strip() for l in f.read().splitlines()]

    for linenum, line in enumerate(content, start=1):
        # Skip empty lines
        if not line:
            reject("empty line", linenum, line)
        else:
            # Skip full-line comments
            if line[0] == '#':
                reject("full-line comment", linenum, line)
            else:
                # Remove in-line comments
                line_no_comments = line.split('#')[0]

                # Split lines at comma and strip whitespace
                splitline = [s.strip() for s in line_no_comments.split(',')]

                # Line only valid if containing at least two entries
                if len(splitline) < 2:
                    reject("not enough entries", linenum, line)
                else:
                    # First valid line is header
                    if first:
                        first = False
                        column1_header = splitline[0]
                        column2_header = splitline[1]
                    else:
                        # Convert first 2 entries to floats and add to respective columns
                        try:
                            entry1 = float(splitline[0])
                            entry2 = float(splitline[1])
                            # col_number.append(linenum) # line number also added for debugging if needed
                            column1.append(entry1)
                            column2.append(entry2)
                        except ValueError:
                            reject("invalid value type", linenum, line)
except FileNotFoundError:
    print(f"Error: The file {file_name} could not be found.")
except Exception as e:
    print(f"Error: {e}")

# Plot data if available
if not (column1 and column2):
    print("No valid data to plot.")
else:
    plt.plot(column1, column2, label=column2_header)
    plt.title(f"{column2_header} vs. {column1_header}")
    plt.xlabel(column1_header)
    plt.ylabel(column2_header)
    plt.tight_layout()
    # plt.legend()
    # plt.grid()

    # Save plot as image
    # plt.savefig("data.png") # Comment or uncomment to save plot as png

    # Display the plot
    plt.show()
```