Introduction to Numerical Methods of Engineering Analysis
University of Florida
Mechanical and Aerospace Engineering

Homework 9: The MATLAB Will Be with You, Always
Due[1]: Wednesday, April 26, 2023, 11:59pm.

Instructions:
1. Submit your solutions as a PDF document on Canvas.
2. ***Include any and all plots in the PDF.***
3. Also submit your individual .m files. You *may not* use MATLAB's built-in integration functions.
4. Do not submit a zipped file.
5. Remember to use a superfluous amount of comments!

Problem 1: Our rover is nearing the end of its service life, and its solar panels are covered in dust. To extend its life, we need to find a high point where the sun will be stronger. The data file `hw9_p1.mat` contains Martian elevation data. Both values are in meters.

Write a MATLAB script called `hw9_p1.m` to compute the Lagrange interpolating polynomial from the data, and plot the data and the interpolated polynomial. Have your script display the maximum elevation and its location (i.e., corresponding x value) in the Command Window.

Note: As usual, remember axis labels and legend, and make sure your curve is smooth.

Problem 2: Write a MATLAB script called `hw9_p2.m` to solve problem 1 via the natural cubic spline. (I.e., compute and plot the natural cubic spline, along with the data points, and have your script display the maximum and its location.)

Note: As usual, remember axis labels and legend, and make sure your curve is smooth.

Problem 3: Let $a = 0$ and $b = 10$. Integrate $f(x) = x^3 + x^2 + x + 1$ in the following ways:

a) analytically,

b) using the trapezoidal rule,

c) using the composite trapezoidal rule with $h = 5$,

d) using Simpson's 1/3 rule,

e) using Simpson's 3/8 rule.

Discuss your results: do they match the theory? (Spoiler: Of course they do. How so?)

[1]Canvas will remain open for submission for a couple of days after that with no late penalty.

Note: You may solve this problem either by hand or using MATLAB. If you solve it by MATLAB, name your MATLAB script `hw9_p3.m`, and have your script display your answers in the Command Window (and remember to use a superfluous number of comments). If you solve it by hand, show your work.

Problem 4: Integrate the function from problem 3 from $a = 0$ to $b = 10$ using Romberg integration (two iterations, i.e., $k = 2$). Use step sizes of $h_1 = 10$ and $h_2 = 5$.

Discuss your results: do they match the theory? (Yes, duh. In what way?)

Note: You may solve this problem either by hand or using MATLAB. If you solve it by MATLAB, name your MATLAB script `hw9_p4.m`, and have your script display your answers in the Command Window (and remember to use a superfluous number of comments). If you solve it by hand, show your work.

Problem 5: Write a MATLAB script called `hw9_p5.m` to use Euler's method and the classical fourth-order Runge-Kutta method to solve the following differential equation from $x = 0$ to $x = 10$:

$$\frac{dy}{dx} = x - y,$$

where $y(0) = 0$ and $h = 2$. Plot the resulting trajectory, $y(x)$ for each method in the same plot. Also plot the analytical solution to the o.d.e., which is of the form

$$y(x) = Ce^{-x} + x - 1,$$

where C is some constant.

(Remember to add a legend and axis labels.)

Discuss your results: which numerical method is better?

Suggestion: Use a function handle.