Andrew Garwood
Written Homework 1

Problem 1:
(a) Each of these variables is an approximation (or exact value) of π. Mathematically, all three are different. What do you see when MATLAB/python displays these numbers? Do they look different?

When MATLAB displays these numbers, they appear the same in the command window when using “format short.” However, when using “format long,” y and z appear the same, while x is different because it did not have as many decimal places defined.

(b) One way to check if two numbers are actually different is to subtract them from each other and see if you get exactly zero. Calculate x − z. Is it exactly zero? Does that surprise you? Why/why not? Now calculate y − z. Is it exactly zero? Does that surprise you? Why/why not?

Calculating x - z, I get x - z = 7.346410206832132e-06. This does not surprise me because x only stores the first 4 decimals of pi, while pi is irrational. However, y - z is exactly zero. This surprised me because the number of decimals in y is finite, while pi *should* have infinite decimals.

(c) Try to figure out how many decimal places MATLAB/python store in pi or np.pi. Explain in your own words how you figured this out. Do you think that this is a problem with how the language stores pi, or is it a more general issue? (You do not have to attach your code for this last part; just explain in a sentence or two how you approached the question.)

MATLAB stores 32 digits total, so it’s 31 decimal places for pi. I figured this out after reviewing researching MATLAB commands regarding digits. Typing “digits” into the command window displays how many digits MATLAB will store for a variable. One can modify this number by providing an argument to the function digits(d) where d is the number of digits you want to store. The constraints on d are that it must be between 2 and 2^29 (according to the error message I received). I do not necessarily think this is a problem with how the language stores pi. Because pi is irrational it is impossible to store all of it - so it is a more general issue of how to store/represent an irrational number.

Problem 2:
fig1:

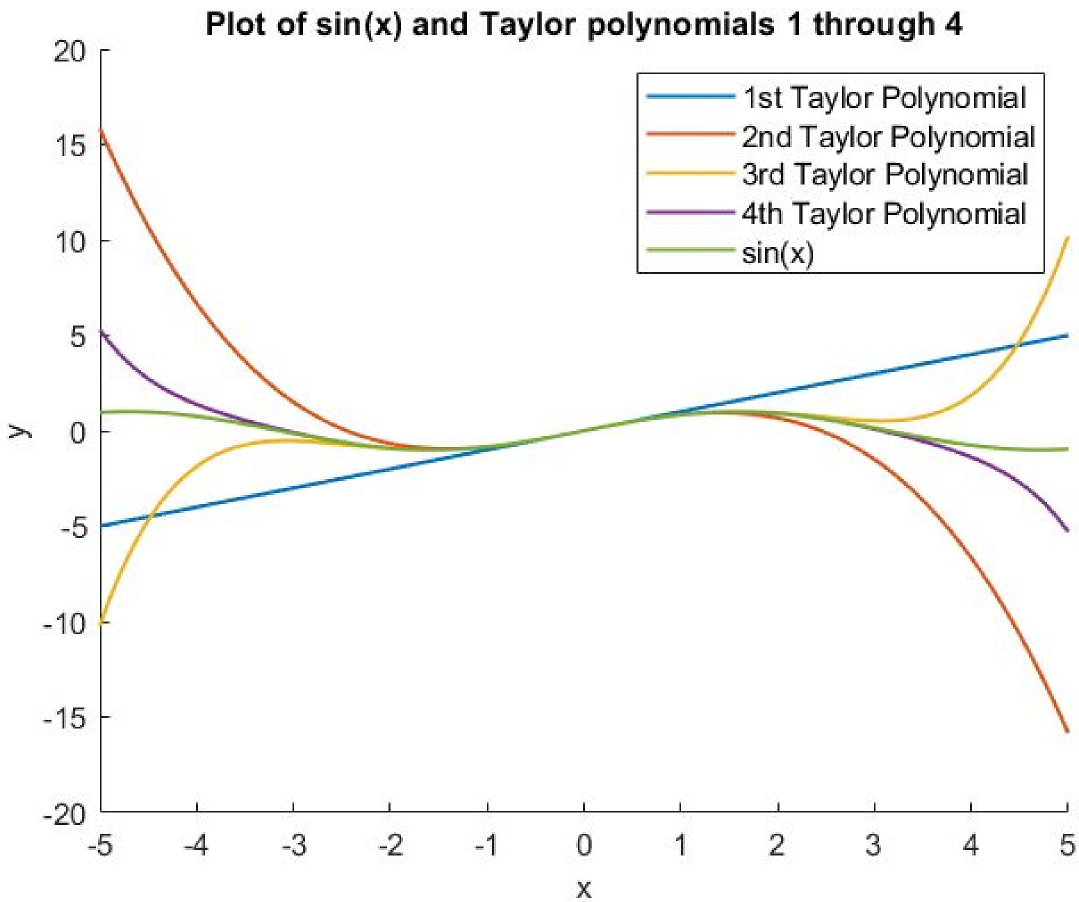

fig2: Code Used to plot the above figure:

C: ▸ Users ▸ drewg ▸ Desktop ▸ AMATH301_MATLAB_PROJECTS ▸

Editor - C:\Users\drewg\Desktop\AMATH301_MATLAB_PROJECTS\garwood_writtenHW_2.m

garwood_codingHW_2.m | garwood_writtenHW_2.m | garwood_writtenHW_0.m

```matlab
format long;
x = 3.1416;
y = 3.14159265358979323846264338327950288419 7;
z = pi;
% digits
% vpa(pi) Variable-Precision Arithmetic


clear all; close all; clc;
x = -5:0.1:5;
s = sin(x);
T1 = x;
T2 = x  - (x.^3) / factorial(3);
T3 = T2 + (x.^5) / factorial(5);
T4 = T3 - (x.^7) / factorial(7);
figure()
hold on
plot(x, T1, 'DisplayName', '1st Taylor Polynomial', 'LineWidth', 1.25); %
plot(x, T2, 'DisplayName', '2nd Taylor Polynomial', 'LineWidth', 1.25); %
plot(x, T3, 'DisplayName', '3rd Taylor Polynomial', 'LineWidth', 1.25); %
plot(x, T4, 'DisplayName', '4th Taylor Polynomial', 'LineWidth', 1.25); %
plot(x, s, 'DisplayName', 'sin(x)', 'LineWidth', 1.25);
% plot(x, y, 'bo') % blue circles on line b = blue o = circle
hold off
xlabel('x')
ylabel('y')
title('Plot of sin(x) and Taylor polynomials 1 through 4')

legend('FontSize', 10)
```

```matlab
clear all; close all; clc;
x = -5:0.1:5;
s = sin(x);
T1 = x;
T2 = x  - (x.^3) / factorial(3);
T3 = T2 + (x.^5) / factorial(5);
T4 = T3 - (x.^7) / factorial(7);
figure()
hold on
plot(x, T1, 'DisplayName', '1st Taylor Polynomial', 'LineWidth', 1.25); %
```

```
plot(x, T2, 'DisplayName', '2nd Taylor Polynomial', 'LineWidth', 1.25); %
plot(x, T3, 'DisplayName', '3rd Taylor Polynomial', 'LineWidth', 1.25); %
plot(x, T4, 'DisplayName', '4th Taylor Polynomial', 'LineWidth', 1.25); %
plot(x, s, 'DisplayName', 'sin(x)', 'LineWidth', 1.25);
% plot(x, y, 'bo') % blue circles on line b = blue o = circle
hold off
xlabel('x')
ylabel('y')
title('Plot of sin(x) and Taylor polynomials 1 through 4')

legend('FontSize', 10)
```