Isaac Hill

Module 5 - Programming Assignment - Regula Falsi

Result Table:

Bracket	Function 1 - y = x^{4}-6x^{3}+12x^{2}-10x+3			Function 2 - y = x^{3}-7x^{2}+15x-9		
	Root	Function Value at Root	# of Ite	Root	Function Value at Root	# of Ite
[2.5, 3.5]	2.999999999999998	7.105427357601002e-15	50	No Root Found	NA	0
[1.5, 2.5]	No Root Found	NA	0	No Root Found	NA	0
[0, 1.5]	1.0027348164706036	-4.085265548070538e-08	100000	1.0000000000000002	0.0	61

Note: I graphed both of the functions and saw that there were two roots at 1 and 3. I saw that the brackets we were supposed to use were [0, 1.5] and [1.5, 2.5]. There wasn't a root in between [1.5, 2.5] but there was at [2.5, 3.5] so I added that bracket to the results table but also kept [1.5, 2.5] to make sure everything was accounted for in the results and analysis.

Analysis:

One of the key things that I noticed looking at the graph of both functions and results, it seems that the amount of iterations needed is related to the curve of the function. Function 1 in the [0, 1.5] bracket had a root of 1.0027348164706036 and had a lower accuracy to the actual answer (1) than the other brackets and functions. When graphing out the function you can see that there is a slow curve in that bracket range for this function. This most likely contributes to the max iterations of 100000 used in the bracket for that function due to the convergence rate depending on the curve. Function 2 in the [0, 1.5] bracket had a root of 1.0000000000000002 and had only 61 iterations to get the answer. It was much more accurate and used less iterations due to the curve being a lot better for the Regula Falsi method. Both functions in bracket [1.5, 2.5] didn't have a root or had any iterations. Function 1 in the [2.5, 3.5] bracket had a root of 2.999999999999998 and only used 50 iterations. The accuracy was very close and had a lower amount of iteration with it being a steep curve. Function 2 in the [2.5, 3.5] bracket didn't have a root but when graphing it out you can see that it passes through y = 0 at x = 3. The program didn't find a root due to both sides being positive unlike the others which had a positive side and a negative side.