NUMERICAL METHODS
ASSIGNMENT 2

SRIRAM AKELLA

1. Problem 1

Let $f(x) = \sin(\cos(\exp(x)))$ whose zeros we wish to find.

1.1. **(a).** We'll find a zero in the interval $(-1, 1)$ using the bisection method. The code I've written for this problem (1a.py) is attached in the .zip file.

At each iteration, the interval is bisected and zeroes in towards the zero of the function. Running this code gives

$$x_{bisection} = 0.4515827052894549, \tag{1}$$

and substituting this back in $f(x)$, my calculator gives 0.

1.2. **(b).** To apply Newton-Raphson, we need the derivative of $f(x)$. Following the hint, I used Mathematica for this and the derivative turns out to be

$$f'(x) = -\exp(x)\cos(\cos(x))\sin(\exp(x)). \tag{2}$$

If we take our initial guess to be $x = -1$, then my code (1b.py – attached in the .zip file) gives me:

$$x_{newton} = 6.082794487111051 \tag{3}$$

as the zero.

1.3. **(c).** The plot for the error is shown in Fig. 1. After a few iterations, the error is almost zero. The code for this is named 1c.py in the .zip file.

1.4. **(d).** The library functions for bisection and Newton-Raphson gives

$$x_{bisection} = 0.451582705291002\,64, \tag{4}$$

$$x_{newton} = 6.082794508951761. \tag{5}$$

The Newton-Raphson method gives a "RuntimeError: Failed to converge after 50 iterations, value is 6.082794508951761."

The reason for this behaviour is the chaotic fluctuations of the chosen function. A plot of $\sin(\cos(\exp(x)))$ in the range $x \in (-1, 7)$ is shown in Fig. 2.

2. Problem 2

In this problem, we'll look at the integral

$$I = \int_0^1 \exp(x)\, dx. \tag{6}$$

2.1. **(a).** The left-point, right-point, and mid-point rules result in

$$I_{left-point} = 1.709618247130625, \tag{7}$$

$$I_{right-point} = 1.726974629236272 \tag{8}$$

$$I_{mid-point} = 1.71827452360616. \tag{9}$$

The code is attached in the .zip file and is named 2a.py. It uses 100 iterations to evaluate the integrals. The analytic integral is $e - 1$, and is about 1.718281828459045. Therefore, the mid-point rule gives the most accurate answer.

Date: January 30, 2022.

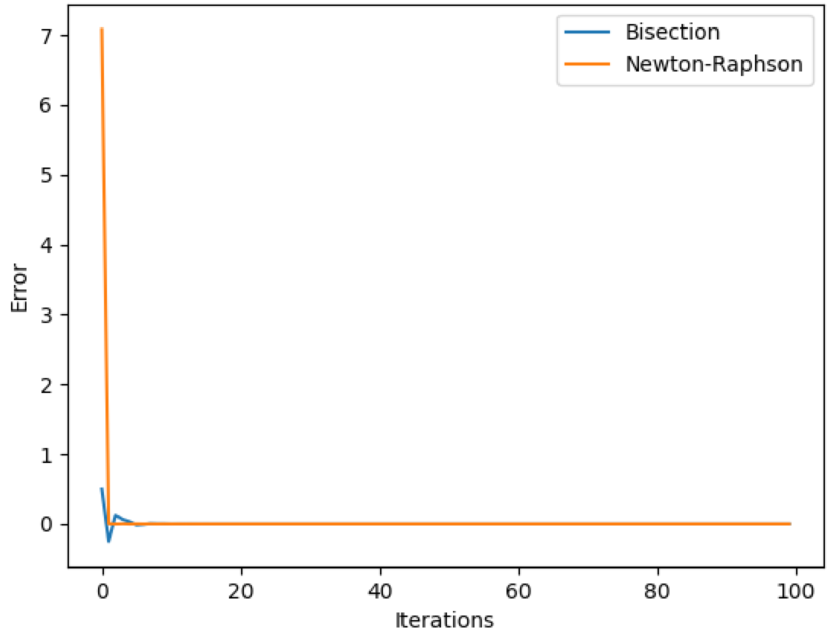

FIGURE 1. The error versus the number of iterations for the Bisection and Newton-Raphson methods. In both methods, the error converges to 0 as the number of iterations are increased.

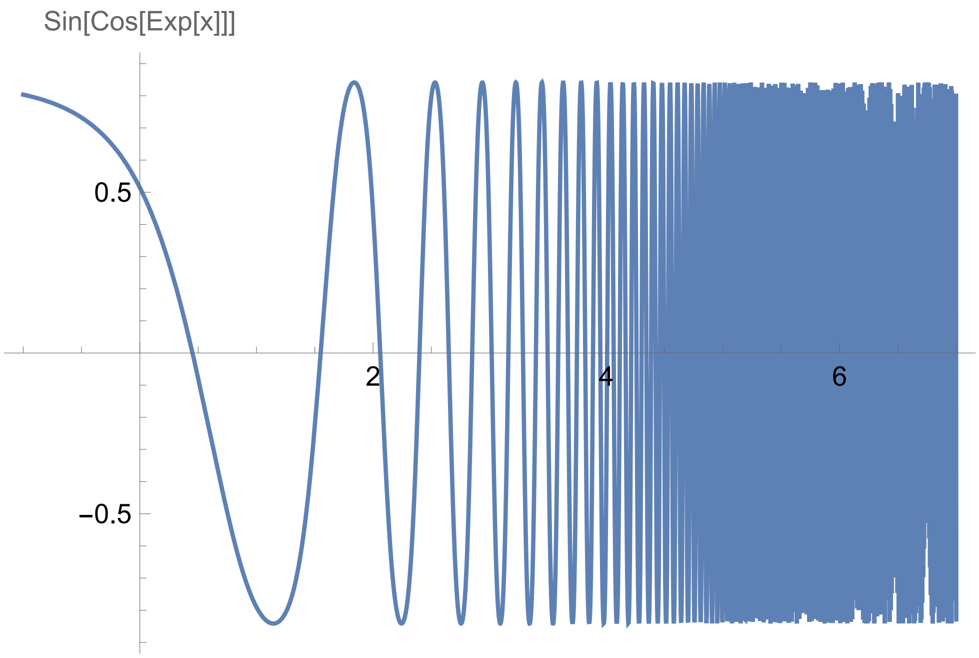

FIGURE 2. The chaotic fluctuations of the function beyond $x = 5$ make almost every number beyond 5 a zero.

2.2. **(b).** Using the Trapezoid and Simpson rules result in

$$I_{trapezoid} = 1.7182961474504177, \tag{10}$$

$$I_{simpson} = 1.7182818285545043. \tag{11}$$

The code for this computation, no prize for guessing, is named 2b.py.

2.3. **(c).** Now we look at the errors in the above methods. The code for this is named 2c.py, and it makes use of library functions. Computing the integral for each iteration without using any library function seemed messy, and the problem did not say anything about not using library functions. Unfortunately, for the left-point, right-point, and mid-point rules, there were no library functions, so I had to do them myself.

The plot with all the errors is Fig. 3.

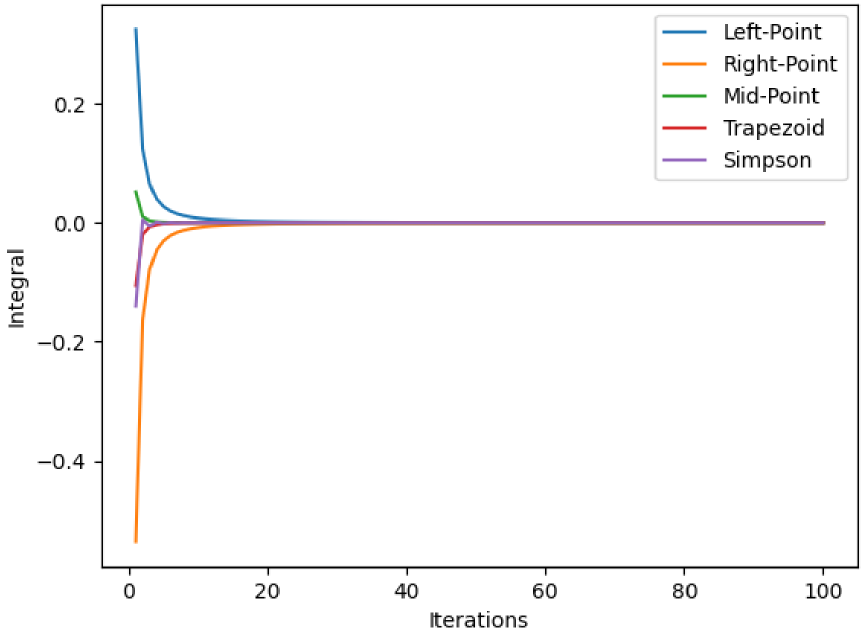

FIGURE 3. The errors in the integration of $\exp(x)$ from 0 to 1 using the various integration techniques.

2.4. **(d).** The code for this problem is named 2d.py. The integrals are:

$$I_{trapezoid} = 1.7182964381834482, \tag{12}$$
$$I_{romberg} = 1.7182818284590782, \tag{13}$$
$$I_{simpson} = 1.7182819878579096, \tag{14}$$
$$I_{fixed_quad} = 1.7182818284583916. \tag{15}$$