LabWork #5
MAT 116E-Advanced Scientific and Engineering Computing (MATLAB)
Instructor: Dr. Ali DEMİRCİ
Teaching Assistant: Gökhan GÖKSU, Göksu ORUÇ Recitation Date: 26/10/2018

Task 1 (Fourier Series)

Write a MATLAB code to find the Fourier series approximation $s(x) = \frac{x}{\pi}$, for $-\pi < x < \pi$ as

$$s(x) \approx s_{approx}(x) = \frac{2}{\pi}\sum_{n=1}^{N}\frac{(-1)^{n+1}}{n}sin(nx),\ for\ \ -\pi < x < \pi$$

REQUIRED

You are asked to complete the following subtasks:

1. Take $N = 1, 5, 10$ terms in the sum. Write a MATLAB program to plot the corresponding approximations. Do not forget to add the **LEGEND**.
2. Take $N = 1, 5, 10$ terms in the sum. Write another MATLAB program to find the sum of squared values (SSE). The formula of SSE is

$$SSE = \sum_{i=1}^{M}(s_i - s_{approx,i})^2$$

where $M = \frac{2\pi}{\Delta x}$ is the number of points whereas s_i and $s_{approx,i}$ are the discrete values of $s(x)$ and $s_{approx}(x)$, respectively.

Task 2 (Babylonian Method to Approximate Square Root)

The first algorithm used for approximating $\sqrt{S}$ is known as the Babylonian method, named after the Babylonians, or "Hero's method", named after the first-century Greek mathematician Hero of Alexandria who gave the first explicit description of the method. It can be derived from (but predates by 16 centuries) Newton's method.

The basic idea is that if x is an overestimate to the square root of a non-negative real number S then $\frac{S}{x}$ will be an underestimate, or vice versa, and so the average of these two numbers may reasonably be expected to provide a better approximation (though the formal proof of that assertion depends on the inequality of arithmetic and geometric means that shows this average is always an overestimate of the square root, as noted in the article on square roots, thus assuring convergence).

REQUIRED

Write a MATLAB **FUNCTION** for the Babylonian algorithm to approximate square root which takes three arguments as input, one of which is the non-negative real number say S, second of which is the initial guess for the square root say x_0 and the other one is the desired accuracy to be achieved say ϵ (absolute error bound). Your algorithm should proceed as follows:

1. Begin with an arbitrary positive starting value x_0.
2. Calculate
$$x_{n+1} = \frac{1}{2}\left(x_n + \frac{S}{x_n}\right).$$
3. Repeat step 2 until the desired accuracy is achieved.

An example to calculate $\sqrt{S}$, where $S = 125348$, to six significant figures, use the rough estimation to three significant figures, use the rough estimation method above to get

$$
\begin{aligned}
x_0 &= 6 \times 10^2 = 600.000 \\
x_1 &= \frac{1}{2}\left(x_0 + \frac{S}{x_0}\right) = \frac{1}{2}\left(600.000 + \frac{125348}{600.000}\right) = 404.457 \\
x_2 &= \frac{1}{2}\left(x_1 + \frac{S}{x_1}\right) = \frac{1}{2}\left(404.457 + \frac{125348}{404.457}\right) = 357.187 \\
x_3 &= \frac{1}{2}\left(x_2 + \frac{S}{x_2}\right) = \frac{1}{2}\left(357.187 + \frac{125348}{357.187}\right) = 354.059 \\
x_4 &= \frac{1}{2}\left(x_3 + \frac{S}{x_3}\right) = \frac{1}{2}\left(354.059 + \frac{125348}{354.059}\right) = 354.045 \\
x_5 &= \frac{1}{2}\left(x_4 + \frac{S}{x_4}\right) = \frac{1}{2}\left(354.045 + \frac{125348}{354.045}\right) = 354.045
\end{aligned}
$$

Therefore, $\sqrt{125348} \approx 354.045$.

<u>**NOTE:**</u> You won't get any credit if there is no <u>**FUNCTION**</u> usage.

Submission Information

Any LabWork submitted after class will be subject to a 20-point deduction per 24 hour period. Extensions should be requested at least 3 days in advance and will only be granted for exceptional reasons (e.g., conference submission). You may work with your friends. Collaboration is strongly recommended. However, each student should be able to present his/her program.