Emile Goulard
uID: u1244855
CS3200 - Assignment 5 - Report
10/17/2022

Question 1

To find the implementation of the 1D Heat Equation using Thermal Diffusivity, open the "HeatEquationClass.m". This file contains the helper code to implement it for Subdivision processes (refer to Question 2).

The methods used to implement the 1D Heat Equation:

- First, I set up the values for alpha, length as provided. To calculate delta x, I assigned it Length/N such that I can get the proper points for any given length. Additionally, the time steps are important for iteration. Therefore, I supplied the length of time to be the "array" the size of the number of steps from 0 to numerical value for final time (which is 5). This is calculated by iterating every delta time step from 0 to 5. Lastly, I had to set the temperature values. The initial temperature is 20, then the starting boundary is 60 and ending boundary is 20. To ensure all values of temperature are being properly counted in the 2D matrix, the temperature array is set with all initial temperature values from 1 to N.
- After setting the values, I needed to iterate in order to solve the 1D Heat Equation for plotting. We start by doing an outer loop from 1 to the length of time steps / delta time (given that the delta time will define how many steps there are). This loop is responsible for setting the boundaries of the Governing Equation at 1 and N because the inner loop will provide those values of the 1D Heat Equation from 2 to N-1. After that, back in the outer loop, we update the temperature value using the formula: Temperature += (delta time * 1D Heat Equation Value). This will be used for plotting.

Question 2

To plot the diffusion of heat for the two different subdivision schemes, open the "A5.m" file and simply run it. Follow the instructions in the README file for more information.

These are the results of the implementation:

Figure 1 (N = 8, 16, 32, 64 and delta time = 0.01s)

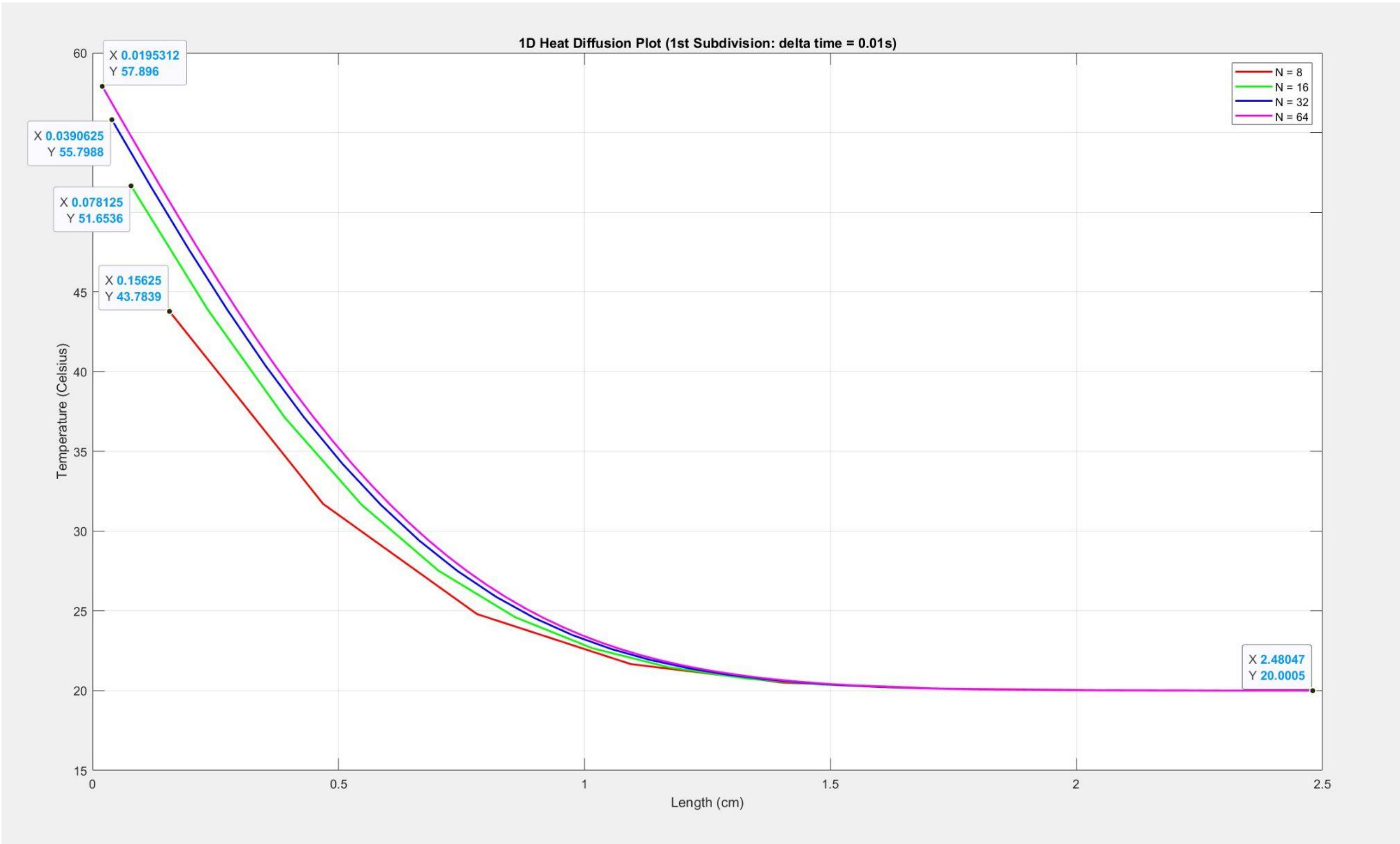
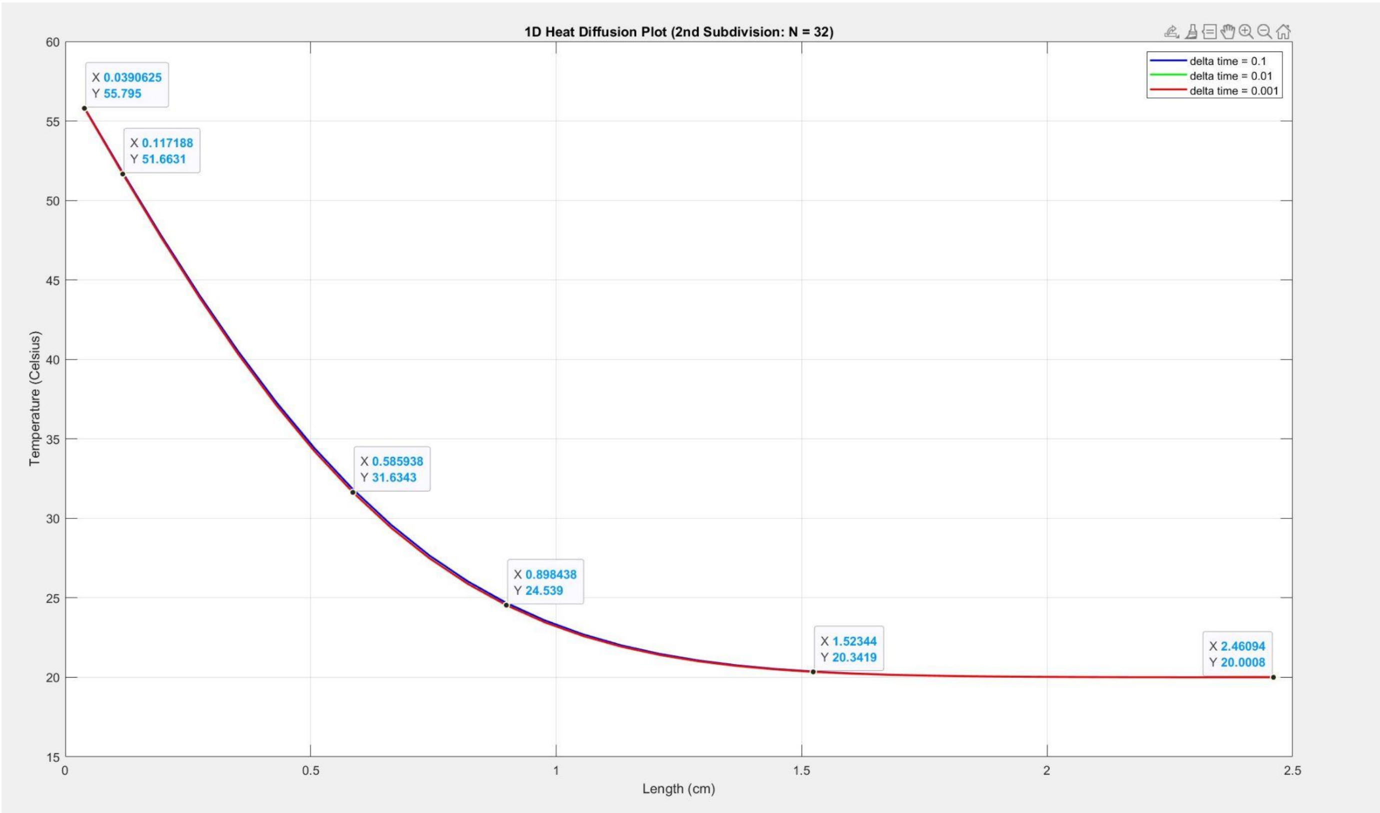
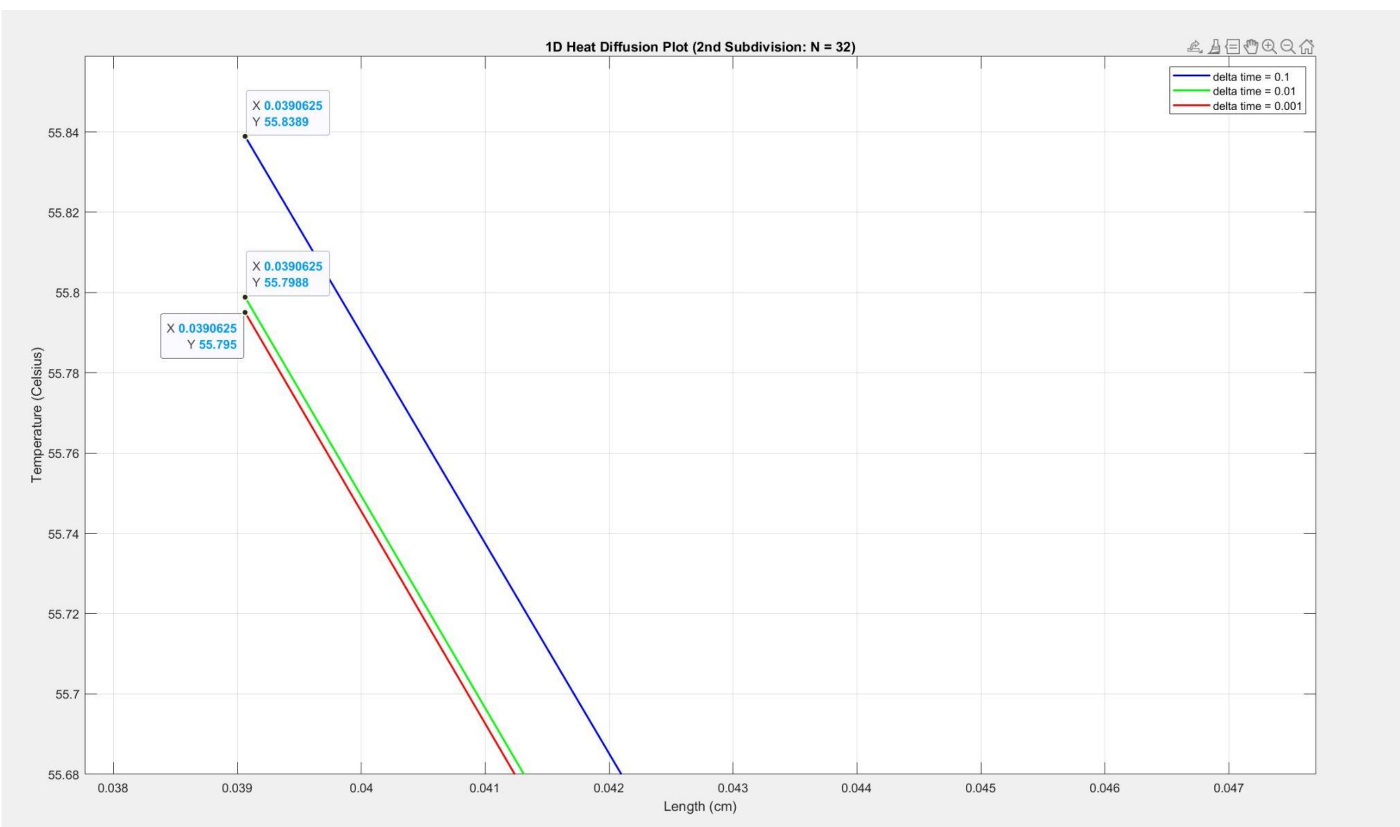
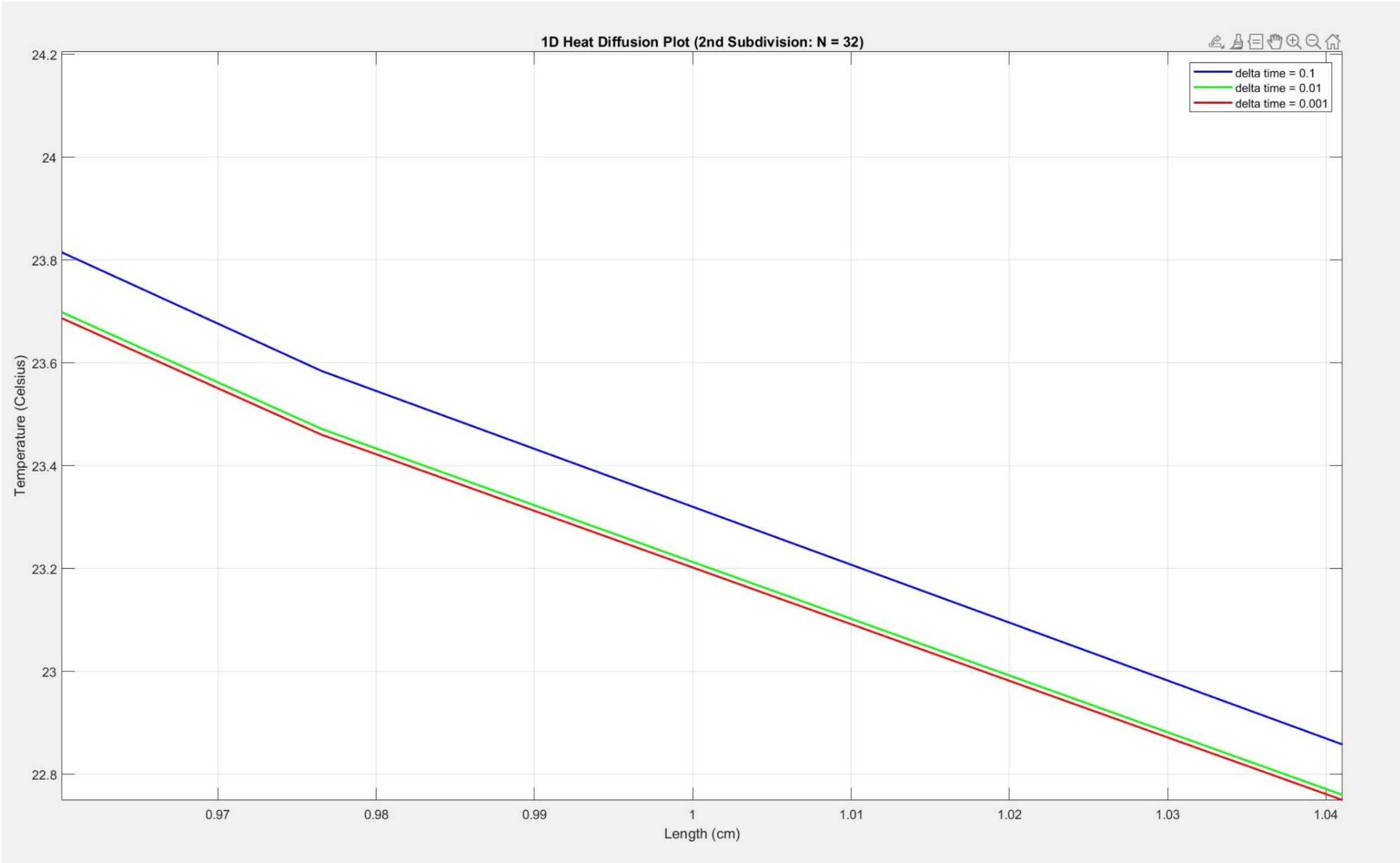


Figure 2 (delta time = 0.1s, 0.01s, 0.001s and N = 32)






1D Heat Diffusion Plot (2nd Subdivision: N = 32)
delta time = 0.1
delta time = 0.01
delta time = 0.001
Temperature (Celsius)
Length (cm)
24.2
24
23.8
23.6
23.4
23.2
23
22.8
0.97
0.98
0.99
1
1.01
1.02
1.03
1.04

Question 3

As you can see in Figure 1, **the results slightly differ for varying subdivisions on that length, or N.** Using delta time as 0.01, all the different subdivision levels eventually converge to 20℃ which is to be expected. For all the N values: N = 8, 16, 32, 64, their initial temperatures are 43.78℃, 51.65℃, 55.80℃, and 57.90℃ respectively. From these initial temperatures, I noticed that the smaller subdivision levels had fewer points along the length because of the fewer number of iterations over the length of time (which is again being iterated by the step size of delta time = 0.01 seconds). This produces a sublinear graph as the temperature steadily decreases. When comparing the values of each N for delta time = 0.01 seconds, the number of subdivisions for smaller N is initially cooler and cools down faster over the length as time increases. This makes sense because, for CPU Processors, the number of subdivisions means more air to ventilate properly and allows for a faster cooling process and slower heating process. Therefore, the number of subdivisions do matter when accounting for the maximum temperature of the CPU as seen by its initial values; the rate at which is cools down is less important but slightly better at smaller N values.

In Figure 2, and its respectively zoomed in graphs, **the results are very minutes for differing delta time values.** Using N = 32 as a basis, all the different time step iterations eventually converge to 20℃ which is to be expected. For all the delta time values: delta time = 0.001s, 0.01s, 0.1s, their initial temperatures are 55.795℃, 55.7988℃, and 55.8389℃ respectively. Again, this plot resembles sublinear convergence as the initial temperatures cool down to 20℃ over the course of 5 seconds (for different time steps). When comparing each time step value for delta time, it only reinforces the information pertaining to N = 32 in Figure 1, except now we see the error margins for larger time steps. Since the delta time, 0.1 seconds, has a larger, slightly more inaccurate, plot of the Temperature over the course of time we can assume that for larger time steps from 0 to 5 seconds will be more inaccurate which makes sense. This is because we can evely distribute the plotted values at the center of the subdivision for much more frequent, yet smaller time step values in order to hone in on the more accurate reading of the Temperature at a given length. While it does take more iterations O(N^2) to complete for smaller delta time values, the values are smaller in error. Overall, it comes down to a person's best judgement when deciding how many time steps to loop through from a certain period of time because longer time periods with really small time steps could make the program take much longer to compute, but far more accurate than my readings in Figure 2.

From these graphs, I can safely assume that the alpha values are critical to figuring out which temperatures are best for the CPU. This is because the alpha value is dependent on variables such as the density, conductivity, and heat capacity because of simply how energy is transferred. It is also very apparent just how drastic the graphs change for different alpha values (which I did on my own time, I don't have graphs to represent those because they aren't needed for this assignment).

Sources

1. https://my.eng.utah.edu/~cs3200/1D_Heat_FD.pdf

2. http://www.cmap.polytechnique.fr/~jingrebeccali/frenchvietnammaster2_files/2017/LectureNotes/heateqn_1d_mit.pdf

3. https://en.wikipedia.org/wiki/Heat_equation