I decided to enhance the .c file to satisfy the data structures and algorithms category because I saw a great opportunity to do more with the data that was being collected by the prototype. My skills are demonstrated by the use of structs to organize related data, the task array used for repeated iteration, and the organization of data analyzing/manipulating functionality into reusable and easily modifiable functions.

The first of the enhancements I made to the thermostat_prototype.c file was the addition of several new functions. To simplify the main function, I moved the logic for board configuration and filling the tasks array into functions that are now called by main. I made the timerCallback function more reusable by replacing the hard-coded number of tasks with the size of the array. Now, if the tasks array grows no changes will need to be made to the task scheduler for it to work properly.

Since I wanted a more accurate way to trigger a reset of all the variables and a dump to the DB at midnight, I created a structure to hold the time. I used a struct for this because the localtime() function returns a single string containing the date and the current local time. I needed the hour and minute in their own int type variables so I could use them in conditional statements. The struct allowed me to return the multiple different data types from the getTime function which was also added during this enhancement. It handles retrieving the date and time, formatting them, and converting the hour and minute before returning it to whatever function requested it. It is used by the tick function detectLastMinute and the dataDump function. The formatting is done via the predefined tm* struct from the time.h file.

While performing this enhancement, I expected to meet the course outcome [CS-499-03], “Design and evaluate computing solutions that solve a given problem using algorithmic principles and computer science practices and standards appropriate to its solution, while

managing the trade-offs involved in design choices." I think that I absolutely met this objective by writing algorithmic, modular code that, in my opinion, is much more reusable, better organized, and more readable than it was prior to being enhanced.

While working on this category, I learned much about the power of using a struct. I have not worked much with the time library (which is something else very valuable that I learned during this enhancement) and the is mainly due to how difficult I have always found it to format and separate the data. Using a struct for this made it so much easier to break the time and date up while keeping it in the same container. I also learned the value of using a structure to return multiple variables from a function in C. That was apparently not something I had encountered before but I am sure to come across again. One challenge that I faced was the localtime function returning an hour earlier than my local time. Fixing this issue was also the only feedback I received on this enhancement. It turned out to be due to the fact that the microcontroller I am using is set to CST whereas my local timezone is EST. I unfortunately concluded that there is no way to change this so I had to settle for adding 1 to the hour except at hour 23 which will result in the hour being set to 00 instead.