Cpre 308 - Joseph Schmidt

Lab 5 Summary:

In this lab we implemented four different ways to schedule processes. These were first come first serve, shortest remaining time, round robin, and round robin priority. These were all scheduling techniques we learned in class and were able to put the theory into practice. First come first serve, involved going through all of the processes and finding the one with the shortest arrival time that was greater than the current time and had not been completed already. Shortest remaining time had similar logic, instead of arrival time though we looked for the shortest remaining time and chose that process. Next was round robin. I implemented this using a global variable that continued to increment and had a real index that was the global variable modded by the NUM_PROCESSES. This was able to keep track of where we left off in the array of processes once we returned from the array. I then looped through the number of processes starting at the global variable until we ran into a process that wasn't finished and that arrived above the current time in the program. The last scheduler was the round robin priority. This added complexity as we had to go through round robin all of the higher priority processes before moving to lower priority processes. I implemented this by going through the number of processes and keeping a max priority process and if any other process had a higher level it would replace it. At the end of the function I then set the global variable to that location + 1 to make sure the same process wasn't getting called twice. Overall, this lab was a very good lab for getting more experience with C and seeing how a scheduler might work in the real world.