**Lab 5 Scheduling – ECE 5780**

Nate Sheffield

A02268057

Nathan Critchfield

A02283426

For this lab we implemented the RMA, EDF and LST scheduling algorithms. With each of these algorithms we saw different but similar schedules created at varying levels of efficiency. Overall, our EDF algorithm performed the best, then LST and lastly RMA but all were fairly close to one another in efficiency. For the RMA algorithm, we implemented a scheduler that sets the priority of the tasks before the simulation time begins and prioritizes periodic tasks over aperiodic tasks. Then once the simulation begins, we calculate what task should be running based on their fixed priorities and will always run the highest priority unblocked task. If there are no periodic tasks unblocked then it will run the highest priority unblocked aperiodic task. In terms of efficiency based on different inputs, we saw that if a set of tasks had a lower utilization, the RMA algorithm would have the same number of preemptions as the EDF or LST algorithms and its aperiodic tasks would have a similar response time.

However, when we ran the RMA algorithm with a task set that was not schedulable without deadline misses, the RMA performed worse than EDF and LST. Using task set 1 (as shown in the table below), RMA had 5 more preemptions than EDF and none of the aperiodic tasks finished before their deadlines. This task showed that the weakness of the RMA algorithm is how it treats aperiodic tasks as second-class tasks and therefore takes longer for them to response and finish executing. Overall, our RMA algorithm performed as explained in class and was implemented using the rules that outline the algorithm.

EDF

LST Non-Strict

In conclusion, we were able to successfully implement and use the three scheduling algorithms to schedule various task sets and compare the differences between schedules they produced. Overall, EDF produced the schedules with the least number of preemptions and deadline misses as well as the shortest response times for aperiodic tasks. While testing task sets with lower utilization the RMA and LST algorithms produced very similar schedules to EDF. However, when using task sets that had higher utilization or were over 100%, we found that EDF produced less preemptions and misses than the other algorithms. Overall, we learned the importance of implementing an optimal scheduling algorithm and to make sure to account for all edge cases of the scheduler to produce the optimal schedule for various task sets.

**Table 1. Schedule Summaries for Scheduling Algorithms based on Different Task Sets**

