Aayahna Herbert & Tiffani Starks
Prof. Carl Martin
CPSC/ECE 3220-002
9 April 2018

Project #2 – Write Up

This project consisted of simulating three different scheduling policies: first in first out (FIFO), shortest job first (SJF), and round robin (RR). FIFO runs based off the phrase, “First come, first serve” by fully executing each task by order of appearance in the work queue; because of this, FIFO doesn’t have to worry about preemption. SJF, on the other hand, does use preemption for it to work properly. Whatever active task is the shortest to execute at that moment, SJF will drop the task at hand and go to the shortest job. The last scheduling policy used was round robin, which switches between every job giving them all equal amount of resources.

The problems we had was in the new_process() function. When running, we initially we were getting the correct values, but the completion and response times were all coming back as -1. We later found that it was because the program wasn’t getting to the main for loop to add the tasks to the work queue. The problem was solved when we realized we were not changing the process status from unloaded (0) to loaded (1).

To execute our code, make sure the text file being used in the program is in the same directory as the code. Afterwards, using gcc -Wall simulation.c and ./a.out will make the program run smoothly. The text file used is the one that was provided to us by the professor.