Alan Padilla Chua

Axp141330

Project 3 - Summary

This project required us to implement the six scheduling algorithms that have been implemented in the history of Operating systems. The algorithms themselves can be either preemptive or non-preemptive. The project required us to implement this project as a object oriented program using classes to encapsulate all methods and abstract the more complex operations in the algorithms. The scheduler class is abstract but also holds a lot of setup methods that all most of the deriving schedulers use with output or just algorithm operation. I made only the one method abstract called schedule which is called from the menu to run the algorithm.

For the non-preemptive algorithms implementation was rather simple. I kept it simple by have a job class that when calling a “run” method just printed out the name of the job at the amount of time that it was required however it also printed spaces to indicate the shift of time the later in the timeline that the job ran. It was easier to just have the jobs do a printing of the spaces and then print their name and finish with a print line than to use the 2-dimensional string matrix that I used for preemptive algorithms.

For preemptive algorithms implementation was definitely a lot more challenging since it required printing on different lines across the timeline. I solved this by having my abstract schedule class have setup methods that allowed it to create a 2-dimensional string array that each job with their designated row number would print to while they ran during the execution of the algorithm. The project description was also rather vague on the order that the jobs will be read from the file so I went ahead and implemented a job sorter class that could be sorted by calling the java collections sort method. Output to the timeline was also shifted by 1 in my implementation because of the way that jobs that arrived at a time would have to be printed at an equivalent time in the matrix.

Overall this project, although told that it would be simpler probably required a lot more thinking for me especially with preemptive algorithms, that required me adding a lot of methods to the scheduler abstract class. I would suggest that the project description have more detail on what to expect from the input file in regards to order of jobs on the document.