Assignment 1 Report

Author: Patrick Fleming
Student Number: C3253586

Results Review

The results of the program are inline with expectations of given tests. The program doesn't consider data sets that don't have a process starting at 0 time but does deal with idle time when a process ends, and another process is yet to arrive. The program badly implements the algorithms as its time complexity and space complexity are very high. The program is poorly designed as every time, I fixed a problem I would realise that if I just rewrote it in a different manner the problem wouldn't exist. Because of this I rewrote the entire assignment 3 times. After the third time I decided that instead of rewriting it would be better to just go with my current design and just fix any problems that I would encounter.

Testing

The testing that was conducted was specifically designed to test only valid data. No tests were conducted for data outside of the scope of the program. Some testing that was done was large numbers of service time and large delays between arrivals.

Performance of Algorithms

All algorithms were implemented using a loop that would loop until all process where complete. This causes the time complexity of the algorithms to be a minimum of all the service time of the processes plus any overhead for dispatching and decisions. When testing the algorithms, the results showed that for different process test sets different algorithms performed better. In general, priority process (PP) and priority round robin (PRR) had the best overall performance not mattering which test set was used. Shortest Process Next (SPN) would work well with tests that had short service times but when testing large amounts of processing would cause waiting times for certain processes to balloon even if that process had a high priority. First come first server (FCFS) is a very fair way to distribute resources but not a very good algorithm if you consider the difference in service times and priorities.

Unexpected Behaviour

The algorithms were poorly thought out so the design of the program reflects that. Because of this there were many unexpected situations, one such problem was sometimes the algorithm didn't print the right process that was being dispatched. Other unexpected behaviour was it printed only one process many times.