Testing cases/Assumptions (How to test the correctness of your program?)

Performance analysis (Discuss and analyze the performance of each scheduling algorithm you have implemented, and this could provide inputs when you design your own algorithms)

Algorithm one (Shortest Remaining Time):

Algorithm two ():

Program set up and execution (How to compile and execute your project? Which special libraries have been included and used in the application? For what reason the application needs those libraries? On which Linux server the application has been tested and what are the results?)

To compile and execute the project. First, download all the .c and .h file together and put it under a same folder. Then, please compile all the files under apollo using gcc -o temp temp.c order.c factory.c day.c for all the c files. Finally, after compiling, use the ./temp coding to run the temp file which is the main program and it will start executing all the functions for the file.

For some of the functions, files will need to be created before executing, like the addBATCH orderBATCH01.dat will need to create a orderBATCH01.dat file and having the addORDER command inside the dat file for executing.

Special libraries like:

We have tested all the codes under apollo Linux server and the results are

Results/graphs/figures discussion

Conclusion

Appendix – source code file(s) and sample outputs of the application.