CMPE322 – PROJECT1 REPORT - Fahri Can Şanlı - 2015400096

1 - Description of The Platform:

When implementing this project, I used Clion on MacBook. I wrote my all code from scratch on Clion Platform. It has gcc compiler so it didn't cause any problem when I checked it on ubuntu. The purpose of checking on ubuntu was to see if there was any problem to fix but my code works correctly on ubuntu using command line.

2 – Description of My Algorithm:

I want to say that I used ifstream to read from the given files. At the first step, I created two struct, one is for process and the other one is for instruction. Then, I read from definition file and add all processes to a process vector. After this step, I read instructions from correct files for each process and add these instructions to the instruction vector of each process.

The second step is the main part which is the cpu scheduler part. My algorithm starts to work by taking first process from the process vector and continues until the ready queue is empty. At the beginning it writes current ready queue to the output file and then starts to process instructions of current process. Processing of instructions of current process continues until current process is finished or next process arrives. When current process is interrupted, it checks three different conditions. If current process is not finished, it is updated in the ready queue. If current process is finished, ready queue is empty, and the arrival time of next process hasn't come yet, then it adds the next process to the ready queue and makes the general time equal to the arrival time of the next process. And otherwise it adds processes that has already come to system at that moment or before. The part above is valid if all processes haven't arrived in ready queue. If all of them are available in the ready queue, my algorithm takes the first one and processes it until it finishes. This goes on until the ready queue is empty. At the final step, my code writes turnaround time and waiting time of each process to the output file.

3 – How to Compile:

My source code and the files that are going to be read must be in the same folder.

g++ -o [executable name] main.cpp (Adding -O2 to the end allows to work faster)

./[executable name]