CMPE322 - Project 2

2020400147 - Muhammet Ali Topcu

Implementation

Program reads the definitions from "definition.txt" file and creates an array of Processes (which is a struct also implemented in the code, includes info about processes process no, priority, arrival time, process type, executed time, start time, end time, waiting time, turnaround time, executed time quantums, last executed at, last executed line, and instruction count). Also, program keeps track of process nos which will be executed via an array called willExecProcesses so that while adding to the ready queue, they will be checked whether they are eligible to add to the ready queue or not. In addition, program reads each instructions length from the file "instructions.txt" and stores them in an array and also reads what the processes will execute from the files "P1.txt", "P2.txt", ..., "P10.txt" and stores them in another array.

Then, program enters an infinite while loop to start the main algorithm. In each pass, it checks whether any remaining unfinished processes exists or not. If exists, it terminates and program finishes. If not, it creates the ready queue. Then, according to the priority of execution, one process is selected to execute. At each execution, program makes necessary updates on the info of the processes. Program works in this way until termination.

Lastly, program outputs the average waiting time and average turnaround time. If they are integers, they are printed as integers, else if they are floating point numbers, they are printed as one digit after dot.

Running the Program

Firstly, note that to be able to run the program, make sure that definition file to read, i.e. "definition.txt", "instructions.txt" file, and "P1.txt", "P2.txt", ..., "P10.txt" are in the same folder with the program, since program reads initial configuration from these files.

To run the program, first run

make

This command will create an executable file called "scheduler". Then, run

./scheduler

will execute the program.

The variable part of the program is the definition file, which should be changed in the parameter of the readDefinition function so that different definition files could be able to run.

Program is tested on Ubuntu version 20.04.6.

Difficulties that I met

Implementing the priority order of the ready queue was the most difficult part to implement for me.