CMSC-312 Operating System Simulator
User Guide

12/01/17

Devin Alexander V00819003

Reid Barden V00720914

John Mitas V00814570

Running the Simulator on Linux and Mac OS

Use SingleSim.jar to run the simulation without multithreading.

Use MultiSim.jar to run the simulation with multithreading.

First, select a scheduler from the drop-down bar on the top left

To use virtual memory, click the “Use Virtual Memory” box and enter a value for VM size and MM size. Example: VM Size:8192 MM Size: 4096. Otherwise do not check the box.

To generate random processes, type the desired number of processes in the box labeled “Number” and then click the “Generate Random Process” button. Example: Number: 20.

To start the simulation, click the “Start” button.

To stop the simulation, click the “Stop” button.

To increase execution speed, move the Execution Speed slider.

To step through the code, type the number of steps desired in the “0000” box to the right of the “Execution Speed” slider and click the “Step” button.

To run a command, type one of the commands listed in the Commands section, in the empty text bar on the bottom left and click the “Enter” button.

To reset the simulator, click the “Reset” button or type reset in the Command line and then click the start button.

Commands

- PROC - shows all unfinished processes in the system and their information. The process information should include: current process state, amount of CPU time needed to complete, amount of CPU time already used, priority (if relevant), number of I/O requests performed.

- MEM - shows the current usage of memory space.

- LOAD - loads a program or job file into the simulator, and will also include the allocation of the program's PCB and memory space.

- EXE - lets the simulation run on its own. The user can also specify the number of cycles to run before pausing. If there are no processes in the ready queue that are waiting to be scheduled, EXE will return to the command interface.

- RESET – allows the user to manually reset the simulator. All unfinished processes are terminated, and the simulator clock returns to zero.

- EXIT – allows the user to end and exit the simulator.

Job Files

proc1.txt – 2 forks

proc2.txt – 2 forks

proc3.txt – 2 forks

proc4.txt – 3 forks

proc5.txt – no forks

proc6.txt – no forks

proc7.txt – no forks

proc8.txt – no forks

Note: There is full functionality on both Linux and Mac OS, but the GUI was designed for and looks best in Mac OS.