OPERATING SYSTEM – LABORATORY 3: SCHEDULING

Student Name: Phong Hai Do. *Student ID: 309047.*

Firstly, for all the case of running (2, 5 or 10 processes), the simulator always read the configuration file line by line, token by token to initialize the parameters for scheduling algorithm, which are number of processes, the mean and standard deviation to randomly generate the CPU time, time for each process run before it blocks for I/O, and how long the simulation should run.

In this assignment:

- All processes run an average of 2000 ms with standard deviation of zero, which means each process has total amount of allowed runtime is 2000 ms.
- All processes execute 500 ms before it blocks for input or output.
- The maximum amount of time that the simulation should run is 10000 ms.

Because the scheduling algorithm is First-Come First Served and scheduling type is Non-preemptive, so the processed will be served sequentially, with no priority until the whole process is completed.

For example, the first process comes, and it will be served immediately, the first process will run for 500 ms until being blocked for I/O. Right after that, the second process will be served, this process also run for 500 ms until being blocked for I/O, right after that the first process will be resumed and served, because it still has 1500 ms remaining time for execution, this will continue until the simulator runs out of time, which is 10000 ms. The simulation serves the new process if and only if one of two processes complete their execution. A process may not complete their execution because the simulator must to stop on time.

The result in case with 2 processes:

Process #	CPU Time	I/O Blocking	CPU Completed	CPU Blocked
0	2000 (ms)	500 (ms)	2000 (ms)	3 times
1	2000 (ms)	500 (ms)	2000 (ms)	3 times

In this case, all two processes have completed their execution on time.

The result in case with 5 processes:

Process #	CPU Time	I/O Blocking	CPU Completed	CPU Blocked
0	2000 (ms)	500 (ms)	2000 (ms)	3 times
1	2000 (ms)	500 (ms)	2000 (ms)	3 times
2	2000 (ms)	500 (ms)	2000 (ms)	3 times
3	2000 (ms)	500 (ms)	2000 (ms)	3 times
4	2000 (ms)	500 (ms)	2000 (ms)	3 times

In this case, process from 0 to 4 have completed their execution. But in the log file, there is no information about completion of process 4, because the scheduling algorithm, the while loop will end right after the time counter reach to the maximum allowed time for simulation.

The result in case with 10 processes:

Process #	CPU Time	I/O Blocking	CPU Completed	CPU Blocked
0	2000 (ms)	500 (ms)	2000 (ms)	3 times
1	2000 (ms)	500 (ms)	2000 (ms)	3 times
2	2000 (ms)	500 (ms)	2000 (ms)	3 times
3	2000 (ms)	500 (ms)	2000 (ms)	3 times
4	2000 (ms)	500 (ms)	1000 (ms)	2 times
5	2000 (ms)	500 (ms)	1000 (ms)	1 time
6	2000 (ms)	500 (ms)	0 (ms)	0 time
7	2000 (ms)	500 (ms)	0 (ms)	0 time
8	2000 (ms)	500 (ms)	0 (ms)	0 time
9	2000 (ms)	500 (ms)	0 (ms)	0 time

In this case, the first four processes have completed their execution, but process 4 and process 5 has not, because the simulator was running out of time. Which means process from 0 to 3 is completed, the process 4 and 5 is served but not completed, the process 6 to 9 is not served.