EOPSY – Task 3
Scheduling

Taimur Rahman
289009

Simulation 1 – `scheduling2Processes.conf`

Configurations

```
// # of Process
numprocess 2

// mean deviation
meandev 2000

// standard deviation
standdev 0

// process    # I/O blocking
process 500
process 500

// duration of the simulation
//in milliseconds
runtime 10000
```

Summary-Processes

```
Process: 0 registered... (2000 500 0)
Process: 0 I/O blocked... (2000 500 500)
Process: 1 registered... (2000 500 0)
Process: 1 I/O blocked... (2000 500 500)
Process: 0 registered... (2000 500 500)
Process: 0 I/O blocked... (2000 500 1000)
Process: 1 registered... (2000 500 500)
Process: 1 I/O blocked... (2000 500 1000)
Process: 0 registered... (2000 500 1000)
Process: 0 I/O blocked... (2000 500 1500)
Process: 1 registered... (2000 500 1000)
Process: 1 I/O blocked... (2000 500 1500)
Process: 0 registered... (2000 500 1500)
Process: 0 completed... (2000 500 2000)
Process: 1 registered... (2000 500 1500)
Process: 1 completed... (2000 500 2000)
```

Summary-Results

```
Scheduling Type: Batch (Nonpreemptive)
Scheduling Name: First-Come First-Served
Simulation Run Time: 4000
Mean: 2000
Standard Deviation: 0
Process #   CPU Time    IO Blocking CPU Completed     CPU Blocked
0           2000 (ms)   500 (ms)    2000 (ms)         3 times
1           2000 (ms)   500 (ms)    2000 (ms)         3 times
```

Conclusion

Processes can only happen one at a time so when one is being executed the other is blocked. Each process runs for 2000ms on average, and 2 processes would take 4000ms. Our simulation running time is 10000ms so its within our range. Our scheduling type is non-preemptive, meaning processes are only executed after the I/O blockade of a previous process. It uses the first-come first-served protocol where once the second process is blocked, we go back to the previous process and this alternates every 500ms.

Simulation 2 – `scheduling5Processes.conf`

Configurations

```
// # of Process
numprocess 5

// mean deviation
meandev 2000

// standard deviation
standdev 0

// process    # I/O blocking
process 500
process 500
process 500
process 500
process 500

// duration of the simulation
//in milliseconds
runtime 10000
```

Summary-Processes

```
Process: 0 registered... (2000 500 0)
Process: 0 I/O blocked... (2000 500 500)
Process: 1 registered... (2000 500 0)
Process: 1 I/O blocked... (2000 500 500)
Process: 0 registered... (2000 500 500)
Process: 0 I/O blocked... (2000 500 1000)
Process: 1 registered... (2000 500 500)
Process: 1 I/O blocked... (2000 500 1000)
Process: 0 registered... (2000 500 1000)
Process: 0 I/O blocked... (2000 500 1500)
Process: 1 registered... (2000 500 1000)
Process: 1 I/O blocked... (2000 500 1500)
Process: 0 registered... (2000 500 1500)
Process: 0 completed... (2000 500 2000)
Process: 1 registered... (2000 500 1500)
Process: 1 completed... (2000 500 2000)
Process: 2 registered... (2000 500 0)
Process: 2 I/O blocked... (2000 500 500)
Process: 3 registered... (2000 500 0)
Process: 3 I/O blocked... (2000 500 500)
Process: 2 registered... (2000 500 500)
Process: 2 I/O blocked... (2000 500 1000)
Process: 3 registered... (2000 500 500)
Process: 3 I/O blocked... (2000 500 1000)
Process: 2 registered... (2000 500 1000)
Process: 2 I/O blocked... (2000 500 1500)
Process: 3 registered... (2000 500 1000)
Process: 3 I/O blocked... (2000 500 1500)
Process: 2 registered... (2000 500 1500)
Process: 2 completed... (2000 500 2000)
Process: 3 registered... (2000 500 1500)
Process: 3 completed... (2000 500 2000)
Process: 4 registered... (2000 500 0)
Process: 4 I/O blocked... (2000 500 500)
Process: 4 registered... (2000 500 500)
Process: 4 I/O blocked... (2000 500 1000)
Process: 4 registered... (2000 500 1000)
Process: 4 I/O blocked... (2000 500 1500)
Process: 4 registered... (2000 500 1500)
```

Summary-Results

```
Scheduling Type: Batch (Nonpreemptive)
Scheduling Name: First-Come First-Served
Simulation Run Time: 10000
Mean: 2000
Standard Deviation: 0
Process #   CPU Time    IO 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
```

Conclusion

Similar as the previous case, except we have 5 processes instead of 2, meaning that it would take a total time of 10000ms to run the simulation (since each process takes 2000ms). This is identical to our run time defined in the configuration file, 10000ms. This is why we don’t see the completion message for process 4 after the last message. The processes are scheduled in pairs, which is why the last process is executed by itself (since 0 and 1 are pairs and 2 and 3 are pairs, but 4 is alone). The blockade is still 500ms.

Simulation 3 – `scheduling10Processes.conf`

Configurations

```
// # of Process
numprocess 10

// mean deivation
meandev 2000

// standard deviation
standdev 0

// process    # I/O blocking
process 500
process 500
process 500
process 500
process 500
process 500
process 500
process 500
process 500
process 500

// duration of the simulation
//in milliseconds
runtime 10000
```

Summary-Processes

```
Process: 0 registered... (2000 500 0)
Process: 0 I/O blocked... (2000 500 500)
Process: 1 registered... (2000 500 0)
Process: 1 I/O blocked... (2000 500 500)
Process: 0 registered... (2000 500 500)
Process: 0 I/O blocked... (2000 500 1000)
Process: 1 registered... (2000 500 500)
Process: 1 I/O blocked... (2000 500 1000)
Process: 0 registered... (2000 500 1000)
Process: 0 I/O blocked... (2000 500 1500)
Process: 1 registered... (2000 500 1000)
Process: 1 I/O blocked... (2000 500 1500)
Process: 0 registered... (2000 500 1500)
Process: 0 completed... (2000 500 2000)
Process: 1 registered... (2000 500 1500)
Process: 1 completed... (2000 500 2000)
Process: 2 registered... (2000 500 0)
Process: 2 I/O blocked... (2000 500 500)
Process: 3 registered... (2000 500 0)
Process: 3 I/O blocked... (2000 500 500)
Process: 2 registered... (2000 500 500)
Process: 2 I/O blocked... (2000 500 1000)
Process: 3 registered... (2000 500 500)
Process: 3 I/O blocked... (2000 500 1000)
Process: 2 registered... (2000 500 1000)
Process: 2 I/O blocked... (2000 500 1500)
Process: 3 registered... (2000 500 1000)
Process: 3 I/O blocked... (2000 500 1500)
Process: 2 registered... (2000 500 1500)
Process: 2 completed... (2000 500 2000)
Process: 3 registered... (2000 500 1500)
Process: 3 completed... (2000 500 2000)
Process: 4 registered... (2000 500 0)
Process: 4 I/O blocked... (2000 500 500)
Process: 5 registered... (2000 500 0)
Process: 5 I/O blocked... (2000 500 500)
Process: 4 registered... (2000 500 500)
Process: 4 I/O blocked... (2000 500 1000)
Process: 5 registered... (2000 500 500)
```

Summary-Results

```
Scheduling Type: Batch (Nonpreemptive)
Scheduling Name: First-Come First-Served
Simulation Run Time: 10000
Mean: 2000
Standard Deviation: 0
Process #   CPU Time    IO 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 times
6           2000 (ms)   500 (ms)    0 (ms)            0 times
7           2000 (ms)   500 (ms)    0 (ms)            0 times
8           2000 (ms)   500 (ms)    0 (ms)            0 times
9           2000 (ms)   500 (ms)    0 (ms)            0 times
```

Conclusion

For this simulation, we have 10 processes altogether, each taking 2000ms (like the previous simulations). The entire simulation would take 20000ms (average), but our range is only 10000ms (like the previous two cases). This is why we only see 6 processes on our summary and only 4 are actually completed. The last 4 processes aren’t even executed. Processes are executed in pairs with the I/O blockade switching after 500ms, so this time there’s process: 4 and process: 5 acting interchangeably. It falls beyond our range (12000ms > 10000ms) so it is incomplete.