**SYSC4001 Assignement 3**

**Rami Ayoub – 101261583**

**Louis Pantazopoulos – 101273477**

**Due Date: Friday, December 6th, 2024**

**Run different simulation scenarios using the different schedulers you defined. Analyze the results obtained**

A simulation of 10 scenarios was conducted to analyze the throughput, average turnaround time, and average wait time for the FCFS, nonprimitive, and RR schedulers. The metrics were compared to processes with the similar I/O and CPU bursts. We began by calculating the throughput (processes per unit time) for each of the 10 simulations in each type of scheduler. The results were gathered in Excel and were the following:

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| FCFC TP | 5/205 | 5/230 | 5/192 | 5/343 | 5/186 | 5/214 | 5/199 | 5/245 | 5/200 | 5/202 |
| NONPR TP | 5/235 | 5/267 | 5/218 | 5/376 | 5/206 | 5/250 | 5/239 | 5/286 | 5/215 | 5/218 |
| RR TP | 5/205 | 5/208 | 5/192 | 5/293 | 5/186 | 5/214 | 5/199 | 5/226 | 5/200 | 5/202 |

Figure 1. Shows the throughput (TP) for each scheduler when running the 10 scenarios.

The non-preemptive scheduling algorithm produces the lowest throughput, and the FCFS and RR scheduling algorithms produce the highest throughput, with RR producing slightly better throughput than FCFS. This can be evident specifically for scenarios where a process has a CPU burst time greater than 100, otherwise it's the same throughput time as FCFS. The average throughput for the 10 scenarios for each scheduling algorithm was calculated as follows:

| FCFC Avg | NONPR AVG | RR Avg |
|---|---|---|
| 0.0232 | 0.0205 | 0.0239 |

Figure 2. The average throughput for the scheduling algorithms from the 10 scenarios

Therefore, it can be concluded that the RR has the best throughput average for the simulation of the 10 scenarios, with FCFS having the second best and non-preemptive having the third best.

Next, the average turnaround time (TAT) and average wait time (WT) were calculated for each of the 10 simulations in each type of scheduler. The results were gathered in Excel and were the following:

| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| FCFC | AVG TAT | 147.8 ns | 167.8 ns | 127.6 ns | 175 ns | 132 ns | 151.8 ns | 122.8 ns | 177.4 ns | 122.6 ns | 151.6 ns |
| NONPR | AVG TAT | 103.2 ns | 114.4 ns | 92.2 ns | 151 ns | 86.6 ns | 115.8 ns | 97.8 ns | 122.6 ns | 90.6 ns | 96.6 ns |
| RR | AVG TAT | 147.8 ns | 163.4 ns | 127.6 ns | 165 ns | 132 ns | 151.8 ns | 122.8 ns | 173.6 ns | 122.6 ns | 151.6 ns |
| | | | | | | | | | | | |
| | FCFC Avg | | NONPR AVG | | RR Avg | | | | | | |
| | 147.64 ns | | 107.8 ns | | 145.82 ns | | | | | | |

Figure 3. Shows the AVG TAT calculated for each of the 10 scenarios for each scheduling algorithm and the Avg of the TAT AVG for each scheduler at the bottom.

It can be concluded that the non-preemptive scheduler has the lowest TAT time, and FCFS and RR have a similar TAT time with a slight difference of RR having a lower TAT for scenarios where the process has a CPU burst time greater than 100. This means that the non-preemptive scheduler completes its task the fastest and RR and FCFS complete their tasks at a similar time where RR is slightly faster.

| | | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| FCFC | AVG WT | 98.8 ns | 118.4 ns | 81 ns | 113.2 ns | 89.8 ns | 99 ns | 76 ns | 124 ns | 78.4 ns | 106.8 ns |
| NONPR | AVG WT | 54.2 ns | 65 ns | 43.7 ns | 89.2 ns | 44.2 ns | 63 ns | 44 ns | 69.2 ns | 46.4 ns | 51.8 ns |
| RR | AVG WT | 98.8 ns | 114 ns | 81 ns | 103.2 ns | 89.9 ns | 99 ns | 76 ns | 120.2 ns | 78.4 ns | 106.8 ns |
| | | | | | | | | | | | |
| | FCFC Avg | | NONPR AVG | | RR Avg | | | | | | |
| | 98.54 ns | | 57.07 ns | | 96.73 ns | | | | | | |

Figure 4. Shows the AVG WT calculated for each of the 10 scenarios for each scheduling algorithm and the Avg of the WT AVG for each scheduler at the bottom.

It can be concluded that the non-preemptive scheduler has the lowest WT time, and FCFS and RR have a similar WT time with a slight difference of RR having a lower WT for scenarios where the process has a CPU burst time greater than 100. This means that the non-preemptive scheduler has tasks that wait the least in the ready queue and the FCFS and RR schedulers have tasks that wait a similar time in the ready queue with RR having a slightly less wait time than FCFS.

**Record the use of memory and analyze the results obtained.**

Lastly, we analyzed the use of memory for the 10 scenarios for each scheduling algorithm. All of the scheduling algorithms used Best-Fit allocation and the average use of memory for the 10 scenarios was calculated as follows:

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| FCFC AVG Memory Used (Mb) | 14.54 | 13.72 | 14.27 | 13.72 | 13.72 | 14.63 | 13.45 | 13.72 | 13.72 | 14.54 |
| NONPR AVG Memory Used (Mb) | 14.81 | 13.19 | 14.27 | 13.72 | 14.81 | 14.54 | 13.45 | 14.81 | 14.54 | 14.81 |
| RR AVG Memory Used (Mb) | 14.54 | 13.72 | 14.27 | 13.72 | 13.72 | 14.54 | 13.45 | 13.72 | 13.72 | 14.54 |
| | | | | | | | | | | |
| | FCFC Avg | | NONPR AVG | | RR Avg | | | | | |
| | 14.003 | | 14.295 | | 13.994 | | | | | |

Figure 5. Shows the average memory use for each scenario for each scheduling algorithm and then the average of each scenario was calculated.

Based on the calculation, the non-preemptive scheduling algorithm uses the most memory on average compared to FCFS and RR which use a similar amount of memory. However, this difference is very minimal and would require multiple tests such as using different allocation strategies and memory sizes to make a conclusion based on which scheduling algorithm utilizes the most and least memory size.