Morgan Bierey Guid: 2508348b Lab group: 6 , Joshua Ekundayo Guid: 2466383e Lab group: 6

Intro – This report covers how the results produced from scheduling algorithms using interesting seeds, deviates from the results produced from the same scheduling algorithms, using standard seeds, such as the one in the sample output (1523376833). The interesting seeds used was 1797410758, 2688744162 and 3399474557. The hypothesis highlighted that using a standard seed the most efficient scheduling algorithm was SRTF followed by SJF, RR and finally FCFS.

Using the First Interesting Seed (seed = 1797410758)

New Order using interesting seed: FCFS , SJF,RR,SRTF (slowest to fastest/most efficient)

When using the first of the three interesting seeds (1797410758) the result showed that SRTF had the shortest average waiting time and shortest turnaround time followed by RR , SJF and finally FCFS.

When comparing the results of the interesting seed with the results produced from the sample output seed i.e() it was found that when using the seed 1797410758 some of the scheduling algorithms worked more efficiently than others.

In this case the RR was found to have both a shorter turnaround time and average waiting time than SJF.

This is due to the fact the service time (CPU burst time) was smaller than the quantum time for five out of ten of the processes that were simulated. Causing 5 of the processes to run to completion first time entering the CPU.

This decreased the waiting time drastically as there were less events that had to be added to the end of the queue to be executed further. To summaries this point the more processes that have a service time less than the quantum time, the shorter the average waiting time and turnaround time for the scheduling algorithm RR.

Additionally, the SJF algorithm ran slower as some processes with long service times arrived earlier than the processes with shorter service time, keeping the processes with a shorter service time waiting to be executed, increasing both the average waiting time and the turnaround time.

Note that the turnaround time increases, as the turnaround time is the departure time – arrival time (if waiting time increases then the departure time increases)

Using the second interesting seed (2688744162)

When using the second of the three interesting seeds (2688744162) the result showed that SRTF had the shortest average waiting time and shortest turnaround time followed by SJF, FCFS and finally RR.

When comparing these results to the result of the standard seed (in sample output), it is seen that the algorithm deviate as RR is now the least efficient algorithm.

This is due to the service time of the processes being greater than the quantum, meaning that processes are added back to the end of the queue more times, increasing the average waiting time and turnaround time. However, this result is not realistic, as the algorithm is tested with a zero-content switch time. Note increasing the content switch time decreases the efficiency of RR and STRF as they will have more content switches (more processes are required to be executed more than once) . The greater the quantum time the less content switches, hence more efficient.

In Addition, another factor is when the FCFS algorithm has a process which has a early arrival time and a long service time, the algorithm will run less efficiently as more processes will wait longer to execute. Increasing the average waiting time.

Using interesting seed 3(3399474557)

When using the third interesting seeds (3399474557) the result showed that SRTF had the shortest average waiting time and shortest turnaround time followed by SJF, RR and finally FCFS.

When comparing these results to the result of the standard seed (in sample output), it is seen that the scheduling algorithms have the expected efficiency apart from the FCFS which performed much worse than expected.

The Average waiting time and turnaround time for FCFS was relatively larger than the other schedulers this is because the first process that arrived had a long service time, which kept the other processes waiting for longer, increasing the average waiting time and average turnaround time.

Furthermore, the fifth process that arrived at the CPU also had a relatively large service time (7.667055271847057) which keep the remaining five process waiting for longer.

Increasing the turnaround time and waiting time

In conclusion, The interesting seeds produced results that deviated from the expected /standard results. The interesting seeds highlights the different flaws within the scheduling algorithms. Overall, the SRTF was the most efficient, however this doesn't consider content switch time which is a pure overhead. Moreover, the interesting seeds conveyed that some scheduling algorithms are more efficient than others in different situations/scenarios, thus explaining why in most systems implement a combination of algorithms.