**Analysis**

1. **Of the four simulated algorithms, which algorithm is the “best” algorithm for CPU-bound processes? Which algorithm is best-suited for I/O-bound processes?**

FCFS SJF SRT RR

Simulation 2: 9788 9788 9788 9788

Simulation 3: 16274 16274 16274 16330

Simulation 4: 78352 76258 77150 77579

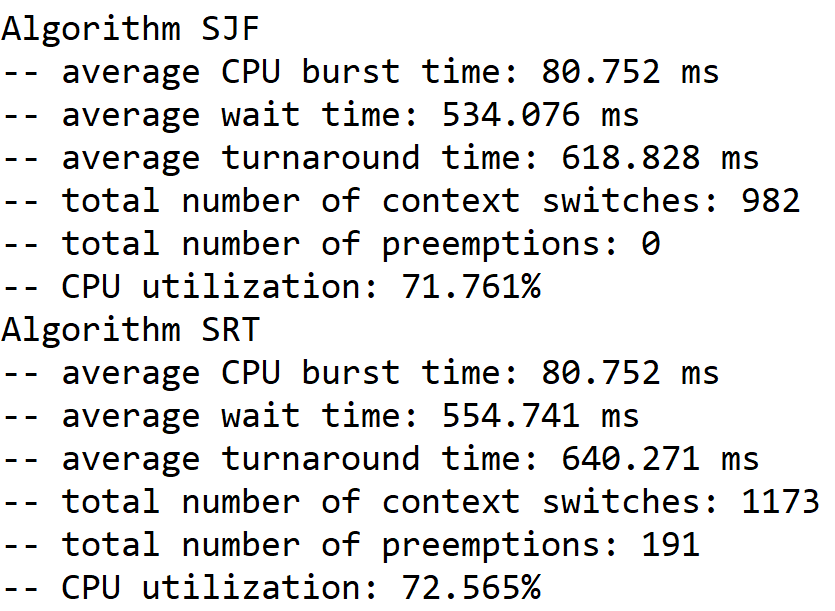
Simulation 5: 78352 76258 77150 77579

1. **For the SJF and SRT algorithms, what value of α produced the “best” results?**

For SJF and SRT the value of α that produces the best results would be when α is approximately 0.5.

1. **For the SJF and SRT algorithms, how does changing from a non-preemptive**

**algorithm to a preemptive algorithm impact your results?**

****

**Figure 3**

./a.out 25 2 0.01 256 4 0.5 64 > output.txt

As shown in Figure 3, changing to a preemptive algorithm increases the number of context switches and increases the average turnaround time. The number of context switches increases because every preemption causes a context switch. The average turnaround time increases for SRT because more context switches occur. Figure 3 also shows that changing to a preemptive algorithm increases the average wait time and increases the cpu utilization.

1. **Describe at least three limitations of your simulation, in particular how the project specifications could be expanded to better model a real-world operating system.**

One limitation of our simulation is that we don’t kill the processes or close any processes. In a real-world operating system, processes could get terminated before they finish all of their bursts.

Another limitation is that our simulation does age any processes. Therefore, our simulation does not have a way to prevent indefinite blocking of processes. A real-world operating system would have aging implemented to prevent indefinite blocking or starvation.

Finally, our implementation is limited by certain variables being constant, such as the timeslice for round-robin. Tuning those variables based on the real-time data of the running processes would better model a real-world operating system.

1. Describe a priority scheduling algorithm of your own design (i.e., how could you calculate priority?). What are its advantages and disadvantages?