**Scheduling Algorithms Analysis**

**Performance:**

We ran a few different sets of processes, with varying cycle sets to get a good feel for the average run times of the various scheduling algorithms.

1. The first set of algorithms we ran generated processes with the number of cycles randomly generated to have a mean of 6000, and a standard deviation of 1000.

Order (least to greatest): SRT, SJF, FIFO, RR, LRT

1. The second set of algorithms we ran generated processes with the number of cycles randomly generated to have a mean of 6000, and a standard deviation of 4000.

Order (least to greatest): SRT, SJF, FIFO, RR, LRT

1. The third set of algorithms we ran generated processes with the number of cycles randomly generated to have a mean of 3000, and a standard deviation of 6000.

Order (least to greatest): SRT, SJF, FIFO, RR, LRT

1. The fourth and last set of algorithms we ran generated processes with the number of cycles all set to 3000.

Order (least to greatest): SRT/SJF/FIFO, RR, LRT

1. Average of the 4 sets.

Order (least to greatest): SRT, SJF, FIFO, RR, LRT

In every case, the SRT algorithm resulted in the shortest average waiting time, followed very closely by SJF. FIFO resulted in the next shortest average waiting time. RR resulted with the second longest average waiting time in every case due to the recurring context switching penalties. LRT had the longest average waiting time due to the recurring context switching when process cycles were near-level resulting in repeated context switches.

**Differences:**