Robert Lowman
Project #2 Review
11/9/2016
CS364: Operating Systems

Algorithm	Throughput	Turnaround	PW Turnaround	Response Time	PW Response Time	Idle Time(MS)	Context Switch(MS)
RR (5ms)	0.419424	6057.5555	4926.421	1453.9714	717.893805	5	10825
RR (25ms)	0.690396	4047.3333	3647.859	937.0857	502.908	33	2375
RR (250ms)	0.804218	5901.8888	6589.52	1413.685	997.23	33	530
FCFS	0.775929	10715.111	11114.596	2651.4	1758.112	631	340
Shortest-Job	0.774327	9077.5556	10760.719	2230.314	1698.628	655	340
Priority (5ms)	0.418877	10971.222	4555.105	2717.57	655.584	33	10825
Priority (25ms)	0.690184	6814	3139.9649	1648.428	417.4513	37	2375
Priority (250ms)	0.80393	6262.3333	3430.474	1506.343	466.038	37	530
LJF (5ms)	0.41534	20838.44	21032.947	5254.85714	3426.18289	271	10770
LJF (25ms)	0.676336	12635.3333	12969.315	3145.314	2070.15	304	2375
LJF (250ms)	0.778681	10850	11239.929	2686.08714	1779.179	400	530
Multiple-Queues	N/A	N/A	N/A	N/A	N/A	N/A	N/A

While the time spent on context switches is larger than most of the others (and the fact that multiple-queues was not working properly enough to test) the Round-Robin algorithm has the best overall statistics for the algorithms tested. This algorithm does not have the best numbers for evey metric, however is surpasses in the most metrics which is why I consider it overall the best. In particular, this algorithm works best with a larger quantum of 250ms as it allows processes to run longer.

The most fair algorithms were the ones that incorporated a quantum and were preemptive. This is because it allows more processes to share the CPU by cutting off longer processes and avoiding processes that get picked last to not be starved of the CPU. While it may lack efficiency and overall speed of how long it takes to finish each process, having a smaller quantum allows these algorithms to be even more fair as there are more context switches and allows more processes to have the CPU.

When the quantum of Round Robin increased, so did the throughput however the turnaround decreased. The quantum being increased also increased the Priority weighted turnaround. The different levels of the quantum did not really matter to response time, except for when a mid-level quantum is used response time is the greatest. Although I was not able to implement Multple Queues correctly, I would assume these statics would be similar for increasing the number of queues as increasing the quantum for Round Robin.

The wildcard algorithm I implemented was longest-job first however it is used preemptively. I found that shortest-job is considerably greater in every metric, except when the quantum is high. My estimate is that when the quantum is increased, it just becomes more like FCFS, which has nothing really much to change with the order.