Zhe Yan (Rcs id: yanz9)

Haotian Zhan (Rcs id: zhanh)

1. (4 points) 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?

RR is the “best” algorithm for CPU-bound processes. Every process gets an equal share of the CPU. Besides, there is no starvation since RR is cycling in nature.

SRT is best-suited for I/O-bound processes. Processes with short CPU burst time have higher priority, which means I/O bursts are prioritized.

1. (4 points) For the SJF and SRT algorithms, what value of α produced the “best” results?
2. (4 points) For the SJF and SRT algorithms, how does changing from a non-preemptive algorithm to a preemptive algorithm impact your results?
3. (6 points) 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.
   1. Limitation 1:
   2. Limitation 2:
   3. Limitation 3:
4. (6 points) Describe a priority scheduling algorithm of your own design (i.e., how could you calculate priority?). What are its advantages and disadvantages?

We can create a new I/O burst First algorithm. In this algorithm, we need to sort the ready queue before the process started. Besides, when a new process arrived, we need to check its I/O burst time with the current running process before entering to the ready queue. If it has less I/O burst time than the I/O burst time of current running process, a preemption occurs. Then, the current running process is added back to ready queue.

Advantage: