Mohamad & Majd Soufan

CS 470 Project 3

Dr. Hwang

3/20/2017

Process Scheduler Discussion

This program satisfies all the requirements included in the project. The program is writing in C# and provides a user friendly GUI. The program contains a Process Scheduler class that keep track of all the processes. The process scheduler is supported with Ready and Wait Queues that contains all the process in the ready or the wait state. It also includes a list that contains all the processes that have been created. The process scheduler reads all the data from an input file. It can analyze and apply all kind of actions like creating a process, interrupts, wait, destroy, event handling and termination. We used the Queues data structure for the Ready and Wait queues. Every time the running process's quantum times decrement to 0, we enqueue the process into the Ready queue and dequeue the last one on the ready queue. Same occurs in the wait event. The process with the specified ID is enqueued into the waiting queue. When the event that is associated with the wait action occurs, the process with the same event ID is dequeued from the Wait queue and enqueued into the Ready queue. The termination operation works recursively. It changes the state of the terminated process and all it's children and their children to terminated. Therefore, it is easy to remove all the processes from the Ready queue, Wait queue, and the processes list.

Questions

1. Small quantum results in more enqueueing and dequeuing from the Ready queue. Every time the quantum time of the running process reaches zero, the process should be enqueued into the Ready queue and the first process on the ready queue should be dequeued and running.
2. The Event handling action because you have to keep track of the running process's quantum time. If it is going to equal 0, the running process should be moved to the ready queue before actually moving the process in the waiting queue to the ready queue.
3. Interrupts.
4. We should have used List instead of actual Queues.
5. Yes. We think this project gave us more insight on RR process scheduling and it's practical application.