Lab 5: Implement a Round Robin Scheduler

Out: October 15th, 2024

Due: At the end of the tutorial (actually by 9:00 p.m. Friday, October 18th, 2024, final deadline).

Marks: 4

Useful Documentation

Provided Files

There is only 1 provided file for this lab `sim.c`. This contains a scheduling simulator built on UBC pthreads (inspired by xv6). You **may** work in groups of at most 4 students.

Description

In this lab, you will be implementing a round robin scheduler with 5 priorities using using multiple queues. It is possible to use a heap, but you are encouraged to use your list library from assignment 1 for the queues.

Inside `set_state()` add processes that are being set to `RUNNABLE` to the queue based on their priority. The priority of a process is inside its process table entry. Reimplement `next_proc()` to return the highest priority runnable process. Treat priority like POSIX does, i.e. a process with a lower priority number has "highest priority".

Deliverables and Grading

A single tar file named `lab5.tar` containing:

- `Makefile` (1 mark)
- `sim.c` (1 mark) modified to run the highest priority process.
- `output.txt` (1 mark) log of the output of the simulator.
- `gitlog.txt` (1 mark)
- Any other necessary source code for compilation.

GENERAL: Ensure that your name, NSID, and student number are at the beginning of every file that you hand in. **If a lab is group work**, all partners' information is required in EVERY FILE. No files with lines more than 80 characters.