Operating Systems - Assignment 1

Andrew Khaz (akhaz)
Arpit Shah (aps180)
Mikhail Soumar (ms2237)

February 27, 2017

Introduction

In this project we implemented our own version of the pthread library. Making use of the yield() function, we created a scheduler that pauses every 50 milliseconds to queue a new task to run from a three-level queue.

1 Specifics

Our three queues are FCFS, with a quantum of 50 ms for the top queue (priority 0), 100 ms for the middle queue (1), and 200 ms for the bottom queue (2). A task that does not fully execute in its assigned quantum gets moved down a queue, where it runs for longer, but the queue is only checked if the top queue has no waiting threads.

1.1 Time Slice Minima

Since itimers are process-wide, each thread is guaranteed to have at least 50 milliseconds multiplied by the time interval (1, 2, 4 for high, middle and low queues) unless they finish before that time and exit, in which case the timer is stopped and reset for the next thread to run.

1.2 Benchmarks

The four examples provided served as excellent benchmarks to determine how well our scheduler works, which is to say, not at all.