OS Project 1

b07902123 蔡奇夆

1. Design

I design two syscalls. syscall(333) for getting time, whereas syscall(334) for printing a string to /dev/kmsg.

I use two CPU cores. One for the scheduler. The other for the child processes.

I use signal to ensure that the child processes start at the right time.

I use sched_setscheduler(pid,SCHED_IDLE,0) to lower the priority of child processes. sched_setscheduler(pid,SCHED_FIFO,99) to make a child process run.

I use a queue to implement FIFO mode.

I use a circular doubly-linked list to implement RR mode.

I use a singular-linked list to implement SJF mode and PSJF mode. The two modes both use an O(N) algorithm to find the job with the shortest finish time.

For more implementation detail, please refer to the source code. It should not be too hard to read.

2. Kernel Version

4.14.25

3. Comparison

I would like to thank @kaienlin (b07902075) for the performance evaluation script. The result is measured by his script.

Please refer to measurement.txt for the comparison between the theoretical and the measured time.

Most of the measured start time and end time are slightly larger than the theoretical time. This may result from the delay in changing the priority of child processes.

I also discover that the loading of the host machine (I run this on a VirtualBox virtual machine) may lead to fluctuations in the measurement. The heavier the loading of the host machine, the more fluctuations in the measurement.