

Operating System Project 1 Report

B04902012 劉瀚聲

B04902032 馬揚格

B04902040 王郁婷

B04902080 徐琮賀

1 Design

1.1 Main Structure

For each process, its attribute (ready time, execution time, start time and process id) is stored in a structure `processData`. A structure `processList` is constructed to maintain a list of `processData`, while processes in it are sorted by ready time.

The scheduler process `S` itself is limited to run on CPU 0 with lowest nice value -20 at the beginning. Once a child process `P` is forked, `P` will limit itself to run on CPU 1, and its nice value is determined by scheduling principle. After finishing setting these property, `P` then execute `./child`, a process that will run million empty iterations for n times, while n is passed through `argv[1]`. To make `P` be able to print its own name, its name is passed through `argv[2]`.

To schedule, `S` idles a process `P1` and awake another process `P2` by setting nice value of `P1` to 19 and setting nice value of `P2` to -20. Child processes won't compete with `S` for CPU resources because they are affined to different CPU.

1.2 FIFO

Maintain two pointer `st` and `ed` pointing to elements in `processList`. They are both pointing the first process at the beginning.

`S` checks if the process pointed by `ed` is ready every time unit. Once the child process is ready, `S` forks it, and `ed` moves right. `S` waits non-blockingly for the

process pointed by **st** every time unit. Once the child process terminates, **st** moves right.

A process is awoken if:

- (1) It is pointed by **S** and has been forked.
- (2) It is forked and **st** and **ed** are pointing the same process.

Actually, the processes between **st** and **ed-1** forms the ready queue.