

OS project1 report

B07902025 陳冠廷

Repo: <https://github.com/timothychen1999/osProject1.git>

設計:

scheduler 上所有的實作在 kernel 上都是用 *SCHED_FIFO* 控制，由於他有允許較高優先度的 process 直接插入(preemptive)，其中 FIFO 模式下優先權是依 ready time 設定，SJF 和 PSJF 皆是按預計執行時間設定，但 SJF 在任意 process 開始執行時就會自行將優先度拉至最高避免被超車，而 RR 中優先權由 scheduler 動態分配，每 500 time units 切換一次

實作上為了省去重新編譯 kernel 的麻煩，寫入核心紀錄的方式是透過 *fprintf* 寫入 */dev/kmsg* (super user privilege)，而取得時間是透過 *clock_gettime* 來得到奈秒級的時間資訊。

實作中 scheduler 和 child 分別被指定(*sched_setaffinity*)在 Logical Core 23 和 24 執行以避免互相干擾。

環境資訊:

Kernel: 5.6.0-rc4

OS: ubuntu 18.04 LTS

Run as superuser

實際情形與理論的比較:

首先，雖然在執行期間兩個核心都穩定被完全占用，但可以發現每個 500 time unit 所需的時間仍不穩定，以及可以注意到在設定完 scheduler 之後有時 process 不會立刻開始執行，另外就是 *fprintf* 和 *fopen* 本身也需要一小段時間執行，因此執行結果和理論有些出入，其中因為 time unit 的不穩定性，這個問題在 round robin 模式下尤為明顯，而在另外三個模式中，為了提升模擬的精確度，在 time unit 執行完執行輸出前會將優先權降至最低以避免耽誤其他 process。