

# OS2020 Project 1 Report

R07922113 資工碩二 李建慶

## 1. 設計

本次作業的設計主要用到的 system call 有兩個，`sys_get_time` 和 `sys_printK`。

分別用來讀取 kernel 的時間和在 `dmesg` 打印 log，畢竟這兩個都接觸到 kernel，需要用特別的 system call 處理，其中在 get time 還用到內核的函數 `getnstimeofday`。

至於處理排程的部分將程式分為三部分：主函式，排程，程序。

主函式負責 IO 和前處理，將策略、資料程序依照名稱、到達時間、執行時間儲存起來，並啟用排程來負責，依照四種不同策略分配。

排程會負責控管，Block 和 Wake up process，依照策略決定更換與否和對象，實作 context switch，同時會 CALL 程序再開始和結束時打印，並在最後得到結束排名。

程序會實作 running 過程，用空迴圈表示實際執行，並打印出完成度，可由此知道整個進程，同時也是用到 system call 的地方，會在程序函式中呼叫 `sys_get_time` 和 `sys_printK`。

## 2. 核心版本

Linux virtualBox 4.14.25, Ubuntu 16.04, x86\_64

### 3. 實際結果與理論結果差異比較

實際跑了程式後發現，某些程式的執行時間依照理論應該要是兩倍，然而實際的時間卻不到兩倍，也有些跑出來的排名結果有些差異，可能是我的 code 中的 time decay 有關，在 while 中，process 會有些許的 decay 用來幫助排程能順利往前，然而這個小的 decay 在每個程序數字差異過大時，可能會使結果在相近答案時產生誤差，便不能得到正確的結束時間。