OS Project1 Report

B07902002 連崇安

設計：

- 前置設定（底下所有的 scheduling policy 都會套用這些設定）：
  把所有的 Process 以陣列儲存，並進行 sort。（先比兩者 ready time 的大小，若相同則比較 execution time 的大小）同時設定 Scheduler（Parent process）和 Process（Child process）會在不同的 CPU 上運作。（Scheduler 在 core 0，Process 在 core 1）
  Scheduler 會用自己的 time clock 來記錄時間，並在某個 Process 的 ready time 到來時 fork 出相對應的 process。
  Process 會用自己的 time clock 來記錄時間，並在 execution time 歸零時呼叫 exit。

  註：此 Project 的 start time 是紀錄<u>行程的產生時間</u>。

- FIFO：
  Scheduler 會在某個 Process 被生成後用 setscheduler 給予 process 遞減的 priority，（第 i 個的 process 的 priority 會被設成 99 – i，也因此本程式有一個限制是最多只能處理 99 個 Process）同時把 setscheduler 的 policy 設成 SCHED_FIFO。

- RR：
  Scheduler 會用一個 queue 來記錄目前正在執行或是就緒的行程，每當有新的行程生成時會把其推入 queue 等待運行。每當程式跑完一個 unit time 時會檢查 queue 的開頭的 process 是否已經 exit，若已經 exit 則把其踢出 queue 並運行下一個行程。當程式跑完 500 個 unit time 時，scheduler 會把正在跑的行程移到 queue 的最尾端，並運行下一個行程。

- SJF：
  Scheduler 會先排出在此 policy 下的行程執行順序（這在知道所有行程的 ready time 和 execution time 的情況下是可能的）。排好順序後，Scheduler 仍會依照原本的順序 spawn process，但行程在生成後會馬上進入休眠，而進入休眠的行程會被 Scheduler 以剛剛排好的順序給一一喚醒並執行，第 i 個行程會在 scheduler 的 timer 超過第 i 個行程的 readytime 以及前 i-1 個行程都執行完成時才會被喚醒。

- PSJF：
  Scheduler 會在新的行程產生時跟現有行程比較剩餘的 execution time 的大小，execution time 較小者會變成執行的對象，較大者會進入一個 list 內並依照剩餘的 execution time 進行 sort。每當有行程結束時，scheduler 會從 list 中抓第一個行程出來繼續跑（如果 list 裡還有行程的話），這個行程在這個 list 裡會有最小的 execution time。

核心版本：4.14.25

討論：

- FIFO：
  執行結果皆符合預期，由於 setscheduler 本身有提供 FIFO 的排程模式，因此在寫這部分的程式和測試時並沒有遇到不符合預期的情形。

- RR：
  大多數的執行結果皆符合預期，唯有少部分的測資（通常是 ready time 或是 execution time 皆為 500 的倍數）有可能會發生完成順序跟理論值不一樣的情形，其原因為在程式的設計上 scheduler 和 process 是使用各自的 time clock 來計時，兩者並沒有同步，因此可能發生 scheduler 認為 time slice 已經達到但 process 卻還沒跑完一個完整的 time slice 或反過來的現象。

- SJF：
  執行結果皆符合預期。雖在程式的設計上，第 i 個行程的喚醒時間是用其等待時間和**前 i－1 個行程的理論執行時間**中取大者，理論上應該計算**前 i－1 個行程的實際執行時間**再做比較。但由於在此排程模式下，setscheduler 會使用 FIFO 的排程模式，且執行的優先度會依喚醒順序遞減，因此能確保第 i 個行程確實會等到前 i-1 個行程都執行完之後才會執行。

- PSJF：
  執行結果皆符合預期。