

OS-Project-1

1. 設計

- FIFO

先來的process先執行，將ready time sort由小到大排序後按順序執行。

- RR (quantum = 500)

將所有process按照ready time由小到大排序。

每個process，若其ready time為當前的時間，則將該process放入一個queue中。

執行一個周期後將remaining execution time減少一個週期，未完成的process 繼續放入queue中。重複動作到所有process完成。

- SJF

選擇執行時間最短的process來執行

將ready time sort由小到大排序，若ready time相同，再按照execution time由小到大排序。

每個process，若其ready time為當前的時間，則將該process放入一個heap中。heap 會依照execution time由小到大排序。

如果目前沒有在執行process，並且heap中有process，則取出execution time最小的process並執行，直到該process執行結束。

- PSJF

每次選擇剩餘執行時間最短的 process 來執行，當新產生的 process 執行時間更短時，就可以插隊。

heap 頂端的process 會隨時與執行中的process比較，若其擁有更小剩餘執行時間，他就會插隊，並把原本執行中的process放回heap。

2. 核心版本

linux 4.14.25 @ubuntu 16.04

3. 比較實際結果與理論結果，並解釋造成差異的原因

順序上執行結果均與理論相同。

但存在有許多誤差，例如執行排程的時間、虛擬機的執行效率、signal傳遞的時間等都影響蠻大的。

在本機上曾經遇到signal傳遞過慢的問題：process結束時，父程序一段時間後才收到SIG_CHLD導致排程亂掉。(process看起來執行的比預定時間久，導致應該先完成的process直到後面process都ready了才完成。在SJF、PSJF有重大影響。)

在換了一台硬體設備叫好後，問題就解決了，排程也正常執行。