

Os proj2 scheduling 10920144 資工三甲 施勝議

1. 開發環境：wsl，使用語言:c++

2. 實作方法流程:

FCFS, HRRN, SJF 皆為使用 non-preemptive

RR SRTF PPRR: 為使用 preemptive

因此這 6 個流程是

Setcurrenttime() -> 依 currenttime setreadyqueue ->從 readyqueue dispatch job 直到結束，而只需要修改排成條件及修改 dispatch 就可以完成。

其中 RR SRTF：還需要修改 setcurrenttime 的條件，currenttime 為可能出現更新的時間點，例如有新的 job 進入，或 job 做完 timeslice。

Pprp 我使用 time 讓 tiime++去觸發該時間點會發生的事，因為 pprp 如果跟上面一樣是抓時間點的話 coding 上會變太複雜。

3. 不同排程法比較:

探討 waiting time:

Waiting Time Average	FCFS	RR	SJF	SRTF	HRRN	PPRR
Input1	14.3	18.4	8.86	8.06	11.6	14.6
Input2	8.4	6.4	8.2	3.75	8.2	9.4
Input3	6.67	11.67	6.67	6.67	6.67	12.5

由上表可以知道 srtf 擁有全部最短的 waiting，這可以透過數學證明來驗證。

由上表知 rr 和 pprp 基本上都會是全部裡面平均最高者，因為所以 job 輪流平均做，也就造成每個 job 的平均等待都很長。

以下是 rr 針對 timeslice 探討:

Input1	1	5	10
Average waiting time	18.3	16.27	14.3

我們可以發現當 timeslice 越大，每個 job 可以做的時間越久，能縮短 waiting time，但 timeslice 越大越接近 fcfs，因此放大 timeslice 只對越早到的 job 越有利。

Fcfs、hrrn、sjf、pprr，基本上哪一個 job 在該排程條件下最有利，該 job 就會在該排程裡最先做完(最有優勢)，fcfs 就是越早到越好，hrrn 就是 ratio 越高越好，sjf 就是 cpubrust 越小越好，pprr 就是 priority 越高越好。

探討 Turnaround Time

Turnaround Time Average	FCFS	RR	SJF	SRTF	HRRN	PPRR
Input1	18.2	22.67	12.73	11.93	15.46	18.53
Input2	13.2	11.2	13	7.8	13	14.2
Input3	24.16	29.16	24.16	24.16	24.16	30

在 Turnaround Time 中，一般情況理論上 rr 會因為需要輪流所以 Turnaround Time 應該會是最大，但這沒有特別意義。

4. 結果與討論:

我認為實務上還是 pprp 或 rr 排程最好，因為不會有像 fcfs 一樣前面一個的 cpubrust 很久，也不會有 stravtion 的狀況(pprr 的話可以增加 agind 機制)，至於其他排程法，由於實務上我們很難知道這個 job 需要多少 cpubrust 所以基本上無法使用。