HW1 Document 資訊三乙 10727219 許志仲

1. 開發環境

平台:Windows 10

使用語言:C/C++

使用Libiary: iostream, fstream, algorithm, string, vector, queue

1. 實作方法和流程
2. 將指定文件之資料全部讀入，先讀method, timeslice，再來讀掉欄位名稱，最後重複讀完所有Process，Process包含ID, Burst, ArrivalTime, Priority四項資訊，存入自訂的名為Process的struct，struct內另外紀錄wait的時間、response ratio，以及兩個大小為5的array方便之後儲存個方法的waiting time和turnaround time，並將其Process push進vector(名為processlist)備用。
3. 依照讀到的method進行排成，method的1~5分別對應FCFS, RR, SRTF, PPRR, HRRN，6則是全部進行一次，各個排成方法會在進行中時印出Gantt chart，並且每個Process執行完之後會回去processlist紀錄waiting time和turnaround time。
4. 所有應該執行的Process執行完之後，依照ID的大小對processlist進行排序，並印出每個Process的waiting time，全部印完之後再印turnaround time。
5. 五種Method的十座細節
6. FCFS
7. RR
8. SRTF
9. PPRR
10. HRRN
11. 分析結果和原因

方法1的結果算是預料之內，畢竟bubblesort的缺點就是資料量一大就會變慢許多，但是方法2 3 4的結果就比較奇怪，方法3雖然較久但是成長幅度較2 4小，理論上multi-process應該最快，研究後發現如果拿掉為了省時兒加上的numba @jit指令，multi-process就會是最快的。最後方法2與4比起來大致相同。