實作作業

程式作業一：

- 寫一個程式可以自檔案輸入各個Process之編號、CPU Burst、Arrival Time、及Priority，並依這些資訊模擬FCFS、RR、Preemptive SJF、Non-Preemptive SJF、Priority等CPU Scheduling。本題必須繪出Gantt Chart，計算每個Process之Turnaround Time及Waiting Time。

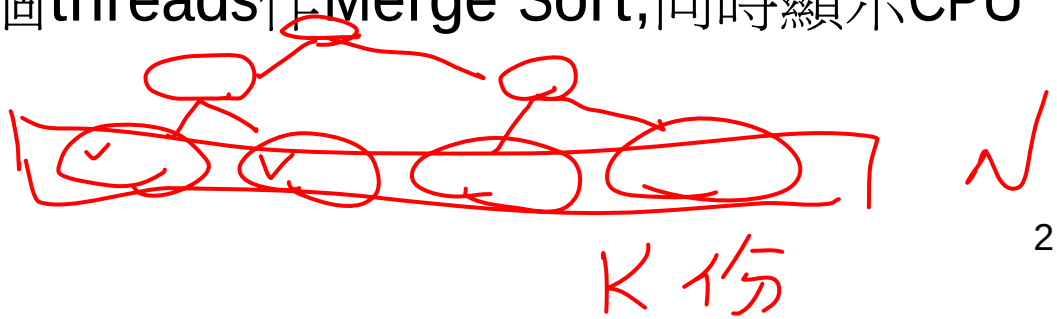
- 程式繳交日為開學後一個月左右，可使用任何電腦語言，任何作業系統均可。

Thread Programming

程式作業二：

假設有一個檔案內有若干個(N個)數目字(至少多於1萬,最多不超過100萬),請寫一個java thread的程式,能夠將該些數目字切成k份(k由使用者自訂),並由K個threads分別進行Bubble Sort之後,再由k-1個threads作Merge Sort,以完成這些數目字之排序.同時顯示CPU執行之時間.本程式需完成下列事項:

1. 將N個數目字直接作Bubble Sort,並顯示CPU執行之時間.
2. 將N個數目字切成k份,並由K個threads分別進行Bubble Sort之後,再由k-1個threads作Merge Sort,同時顯示CPU執行之時間.



Thread Programming

3. 將N個數目字切成k份,並由K個Processes分別進行Bubble Sort之後,再由k-1個Processes作Merge Sort,同時顯示CPU執行之時間.
4. 將N個數目字切成k份,在一個Process內對K份資料進行Bubble Sort之後,再用同一個Process作Merge Sort,同時顯示CPU執行之時間.

Page Replacement

程式作業三：

寫一個程式可以自檔案輸入Page Frame個數及各個Page Reference的次序，並依這些資訊模擬FIFO、LRU、Additional Reference Bits(Shift Register=16 bits ， 每隔5 pages shift一次)、Second Chance 、Least Frequently Used、Most Frequently Used等Page Replacement。本題必須計算每種Page Replacement之Page Fault及Page Replace的次數。 可使用任何電腦語言，任何作業系統均可。

檔案內容規定:第一個Row放Page Frame個數

第二個Row放各個Page Reference的次序

輸入檔定義

Method	time slice		
ProcessID	CPUBurst	arrival time	Priority
.	.	.	.
.	.	.	.
.	.	.	.

- Integer

Method定義

1. FCFS (First Come First Serve)
2. RR (Round Robin)
3. PSJF (Preemptive Shortest Job First)
4. NSJF (Non-preemptive Shortest Job First)
5. PP (Preemptive Priority)
6. ALL

FCFS之處理

- 依arrival time先後次序處理
- 若arrival time相同時, 則依ProcessID由小至大依序處理。

RR之處理

- 先依arrival time先後次序處理, 時候未到的Process不能Run。
- 若arrival time相同時, 則依ProcessID由小至大依序處理。
- Time out時, 若有新來的Process, 則讓新來的Process排在前面。
- 某個ProcessTime slice未用完就結束時, 必須讓下一個Process執行。

PSJF之處理

- 由剩餘CPU Burst最小的Process先處理。
- 若剩餘的CPU Burst相同, 讓沒有用過CPU的先使用, 無法分別時則依arrival time小的先處理。
- 若剩餘CPU Burst相同且arrival time相同, 則依ProcessID由小至大依序處理。

NSJF之處理

- 由CPU Burst最小的Process先處理
- 若CPU Burst最少的Process不只一個, 則依arrival time小的先處理
- 若CPU Burst及arrival time相同, 則依ProcessID由小至大依序處理。

PP之處理

- Priority number小的代表Priority大
- 依Priority由大致小依序處理
- 若Priority相同,讓沒有用過CPU的先使用,無法分別時則依arrival time小的先處理
- 若Priority及arrival time均相同, 則依ProcessID由小至大依序處理

輸出之規定

- Turn around time 及Waiting time
  - 同課本及講義
- Gantt chart
  - 可用圖解或文字表示
  - 文字表示方式

ProcessID	Time	ProcessID	Time	…
P15	3	P62	2	…

  - 圖解