

CS305 作業系統概論 Prog. #3 Real-time Process Scheduling

2018.05.14

一、作業目的

在本作業中，利用模擬的方式來熟悉 Real-time Process Scheduling 的觀念。

二、作業內容

撰寫一個模擬程式自檔案中讀取 Real-time process 資訊，這些 process 的定義為 $t(r, C, D, T)$ 其中 r is the arrival time (ms), C is the CPU burst (ms), D is the deadline, and T is the period. 模擬程式會顯示這些 Real-time process 的使用 rate monotonic scheduling (RMS) 以及 earliest deadline first (EDF) 排程過程。並依照指定的時段輸出模擬過程。在命令列以參數形式，內容是 “rms” 或是 “edf” 來指定是哪一種演算法。

假設作業系統在每一個 ms 來檢查目前正在執行的工作，來判定該工作是否發生 deadline miss，如果在過程中發現某 process 會 miss deadline，程式必須印出訊息，同時模擬中止，此時印出所有發生 deadline miss 的工作。如果在同一時間中有多個 processes 一起到達，則先顯示到達的情況再來顯示排程選擇的進行。如果有 rate 相同的 processes，則依照 pid 大小決定排程優先權：pid 小的 process 具有高的排程優先權。

輸入檔案的範例：

```
# Simulation output time: start end
0 300
# Process info: pid, r, C, D, T
1 0 3 9 9
2 0 4 10 10
3 0 5 15 15
```

例如下面是可能的輸出：

```
> rtsim rms data.txt
0 p1 arrives
0 p2 arrives
0 p3 arrives
0 p1 starts
3 p1 ends
3 p2 starts
...
16 p3 misses the deadline
...
```

三、作業要點

1. 請注意，本作業使用的程式語言是 C/C++，測試平台的作業系統：Ubuntu 17.10 LTS 64-bit。使用的編譯程式為 gcc/g++ 編譯器：7.2。其他平台或程式語言不在本次作業考慮範圍之內。如在測試平台上無法編譯與執行，都不予給分。
2. 模擬的進行方式如下：
 - (a) 工作排程都是由 0 ms 開始模擬。
 - (b) 作業系統在每一個 ms 來檢查目前正在執行的工作是否發生 deadline miss。並且只檢查正在執行的工作。
 - (c) 當某 process 發生 deadline miss 時，模擬中止，同時程式必須印出訊息。但此時如果有多個工作發生 deadline miss，則需要印出所有發生 deadline miss 的工作。

- (d) 資料檔案中，“#”表示註解，第1個非註解行是顯示模擬結果的起始時間與結束時間。接下來n個非註解行是process資訊，執行檔名一律都是rtsim。程式執行必須使用下面方式：

```
rtsim rms data.txt
```

3. 本作業的評分方式如下：

- 能順利以RMS排程 $2 \leq n \leq 5$ 個 processes，可輸出從0~300 ms的模擬結果。此項目最多得40分。
- 能順利以EDF排程 $2 \leq n \leq 5$ 個 processes，可輸出從0~300 ms的模擬結果。此項目最多得40分。
- 可隨意顯示0~300 ms 其中任何一段時間兩個演算法的排程模擬結果。此項目最多得20分。
- 完成以上所有基本功能者，才可按照以下項目，多得其他的分數。如果前面基本功能沒有全部成功完成，以下進階功能不予計分。

如有實作進階功能的同學，請多加一個0或1的參數表示會不會有mutex的考慮。如果沒有指明，預設是0。在文件中要明顯標明程式是否實作進階功能。

例如：

```
rtsim rms data.txt 0      // 表示模擬最單純的RMS
rtsim edf data.txt 1      // 表示模擬考慮priority inversion的EDF
rtsim edf data.txt        // 表示模擬最單純的EDF
```

- 在RMS中加入Priority inversion的情況。假設所有的processes都共用一個mutex，在模擬過程中，每個process一開始執行，就會隨機決定該次執行是否會需要得到該mutex，同時在該次執行結束時，會釋放該mutex。模擬時，需要將這些情況都顯示出來。此項目最多得25分。

例如，一個可能的輸出：

```
...
60 p3 starts, gets mutex
62 p1 starts, gets mutex, blocked
...
```

- 如項目i，但是針對EDF來完成。此項目最多得25分。

4. 本作業需繳交檔案：

- 說明報告：檔案為docx或pdf格式。
 - 報告中必須說明程式的設計理念、程式如何編譯，以及如何操作。
 - 報告中同時必須詳細說明你完成哪些部份。如有用到特殊程式庫，請務必說明。
 - 請務必讓助教明白如何編譯及測試你的程式。助教如果無法編譯或測試，會寄信（最多兩次）通知你來說明，但每說明一次，助教會少給你10分。
- 完整原始程式碼。不可含執行檔。助教會重新編譯你們的程式。

5. 所有相關檔案，例如報告檔、程式檔、參考資料等，請壓縮成一個壓縮檔（不可超過2MB）後上傳至portal。請注意，不可抄襲。助教不會區分何者為原始版本，被判定抄襲者，一律0分。

四、繳交方式：

1. 最終繳交時間：

- 電子檔在 2018.06.06以前，上傳至個人portal。如有多個檔案，將所有檔案壓縮成zip（rar 亦可）格式，然後上傳。
- 上傳檔名格式：「學號_作業號碼.doc」或「學號_作業號碼.rar」。例如：912233_01.doc 或 912233_01.rar。

2. 如有違規事項者，依照課程規定處理。

3. 如需請假，請上portal請假，並持相關證明文件，在請假結束後的第一次上課時完成請假手續，並在一週內完成補交。補交作業將以8折計算。
4. 老師不接受「門縫」方式繳交，助教也不接受任何作業。

五、 如有未盡事宜，將在個人portal板面公告通知。

六、 If you need **any assistance in English**, please contact Prof. Yang.

七、 參考資料

1. 課本第5章
2. www.twwiki.com/wiki/RMS
3. <https://github.com/sigsegved/RTOS-Scheduling-algo-simulation> (此網站資料僅供參考，請勿抄襲)