

CS305 作業系統概論 Prog. #2 Multithreading

2021.10.25

一、作業目的

熟悉如何使用Pthreads的API，撰寫multithreaded program，以及threads彼此之間如何共享資源。

二、作業內容

1. 由main thread 從命令列讀入一個資料檔名，然後從檔案中讀入一個含有括號的 infix 四則運算式。Main thread要印出infix內容，然後連續產生兩個 work threads進行運算。
 - a. Work thread 1 將infix 運算式轉成 prefix 運算式並計算結果。算完後，work thread 1將 prefix 運算式、結果、及整個執行時間印出，並將執行時間傳給main thread。
 - b. Work thread 2 將infix 運算式轉成 postfix 運算式並計算結果，算完後，work thread 2 將 postfix 運算式、結果、及整個執行時間印出，並將執行時間也傳給 main thread。
 - c. 最後由 main thread 印出兩個 work threads 的 id 以及它們所轉換與計算時間。Main thread 並印出整個main thread 的執行時間。
2. 相關資料的說明如下：
 - a. 輸入的infix不會有空格。檔案內只有一個infix算式。數字範圍為正整數（unsigned long long）。
 - b. 輸出的prefix 與 postfix 式子的運算子與運算元以一個空格分開。運算結果如有小數，四捨五入進位為整數。
 - c. 同一個 precedence 等級的運算子，其associativity 關係如 C 語言所定義，例如“+”，“-”具有left-to-right的associativity， $1+2+3 \rightarrow (1+2)+3$ 。
 - d. 執行時間是實際時間(real time)，是指從進入函式開始執行到運行終止所用的時鐘時間(Wall time)。單位為 ms。

例如，假設子執行緒1的TID為2389，子執行緒2的TID為2395，程式執行的可能情況如下：

```
>prog2 prog2data
The infix input: 1+2*4+(7-5)/2
[Child 1] ++ 1 * 2 4 / - 7 5 2 = 10 20ms
[Child 2] 1 2 4 * + 7 5 - 2 / + = 10 22ms
[Child 1 tid=2389] 20ms
[Child 2 tid=2395] 22ms
[Main thread] 28ms
```

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 21.04 64-bit。使用的編譯程式為gcc/g++ 編譯器：10.3。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. 請注意，本作業一定要用Pthread API來進行。任何不用Pthread API的程式，都不予給分。
3. 本作業的評分方式如下：
 - a. 基本功能：依照下面項目的完成程度來給分，如果只能完成部份，將部份給分。
 - i. 主執行緒從命令列讀入檔名，並完成內容輸出。本項滿分10分。
 - ii. 主執行緒能正確產生子執行緒。本項滿分10分。
 - iii. 在不考慮大數運算下，子執行緒能夠正確完成正確的 prefix 與 postfix 的轉換與運算，並印出結果。本項滿分40分。

- iv. 在不考慮大數運算下，子執行緒能夠正確處理infix運算式中的各種錯誤。例如“1+2+”，應輸出錯誤訊息。本項滿分20分。
- v. 主執行緒與子執行緒不透過任何IPC機制交換資料，並且正確印出子執行緒的tid與執行時間。本項滿分20分。
- b. 進階功能：完成以上基本功能者，才可實作以下項目來得分。未完成基本功能者，進階功能的實作不予考慮給分。
 - i. 考慮大數運算，數字最多到25位整數。輸入仍然在正整數範圍內。各種infix算式錯誤都可以處理。本項滿分20分。
- 4. 本作業需繳交檔案：
 - a. 說明報告：檔案為docx或pdf格式。
 - i. 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**。
 - ii. 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。
 - iii. 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**。
 - b. 完整原始程式碼檔案（.c 或 .cpp）。**不可含執行檔。助教會重新編譯你們的程式。請注意**：不可用.txt檔或是.docx檔等非正常方式繳交程式碼，如有類似情形，**助教也會扣10分**。
 - c. **不可以含有病毒，如果含有病毒等惡意程式，本作業0分**。
- 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：
 - a. 電子檔在 2021.11.25 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。
 - b. 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 **an English version** of this assignment or **any assistance in English**, please contact Prof. Yang.

七、參考資料

- 1. 參考課本圖 4.11。
- 2. PThread: <https://computing.llnl.gov/tutorials/pthreads/>
- 3. POSIX 線程 (pthread) 入門文章分享: <http://dragonspring.pixnet.net/blog/post/32963482-posix%E7%B7%9A%E7%A8%8B%28pthread%29%E5%85%A5%E9%96%80%E6%96%87%E7%AB%A0%E5%88%86%E4%BA%AB>
- 4. CPU時間的說明，<http://antoniohsu.blogspot.tw/2011/04/time.html>
 - a. 三種時間測量
 - 1) 實際時間(real time): 從command命令列開始執行到運行終止的消逝時間；

- 2) 使用者CPU時間(user CPU time): 命令執行完成花費的用戶CPU時間，即命令在用戶態中執行時間總和；
 - 3) 系統CPU時間(system CPU time): 命令執行完成花費的系統CPU時間，即命令在核心態中執行時間總和。
- b. CPU時間 = 使用者CPU時間 + 系統CPU時間
5. C/C++ 語言測量時間函數，評估程式執行效能方法整理：<https://blog.gtwang.org/programming/measure-the-execution-time-in-c-language/>