**Operating System: Project 2**

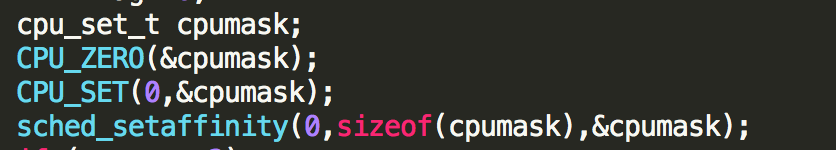
資工二 B04902051 林承豫

資工二 B04902053 鄭淵仁

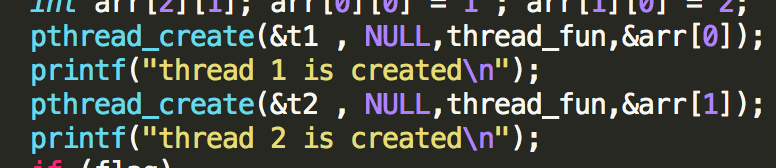
**Part I**

* **Implementation Details**

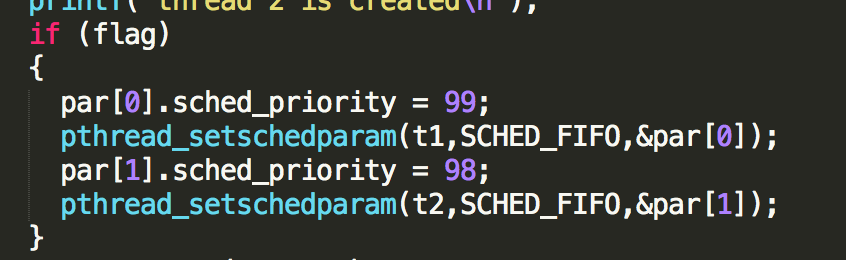
1. Set the number of CPU to one.



1. Create two threads.

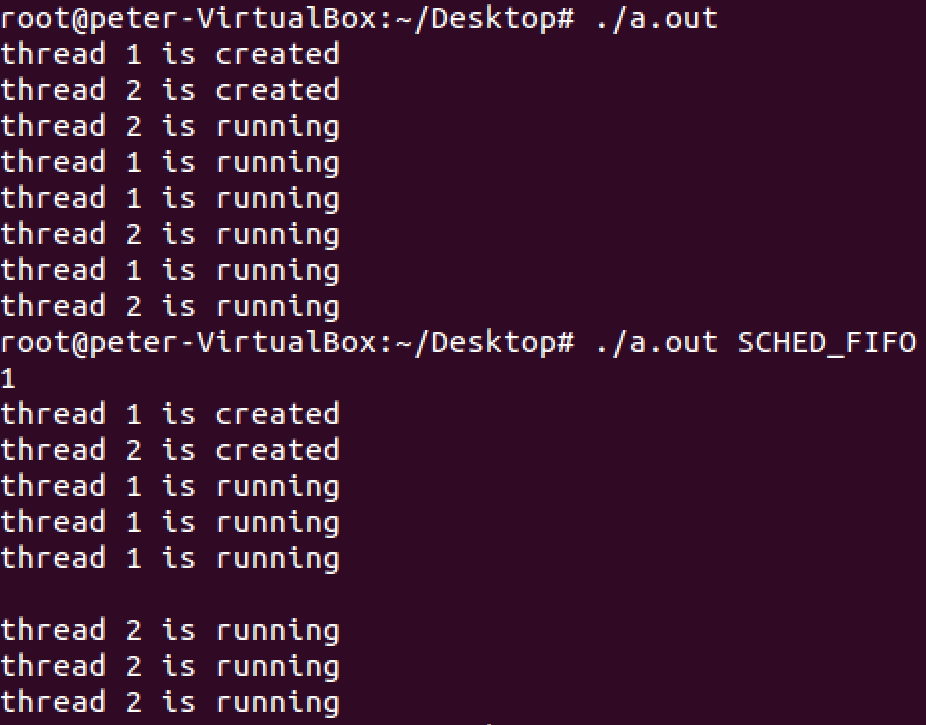


1. If the argument FIFO is input, set the first thread with high priority.  
   (“flag = 1” means program should be run in FIFO)



* **Results**

在Linux裡面實際執行結果如下：

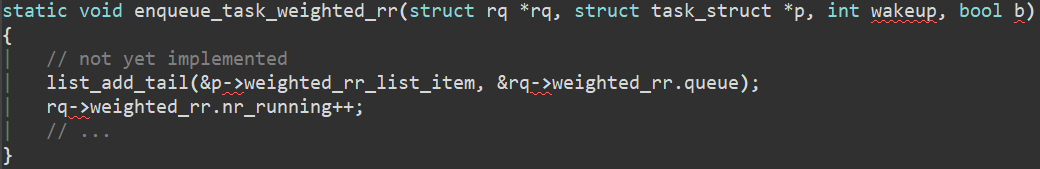


**Part II**

* **Implementation Details**

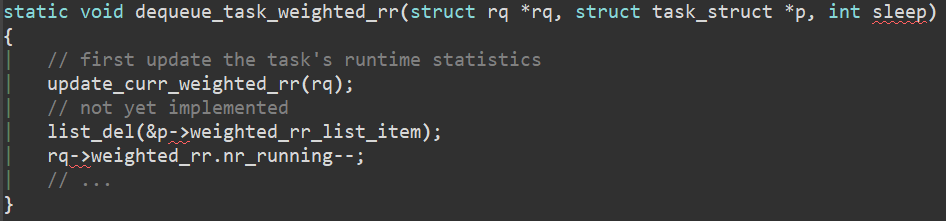
1. **enqueue\_task\_weighted\_rr()**

把task用list\_add\_tail加到ready queue的最尾端，並把ready queue中的數量加一。程式碼如下圖所示：



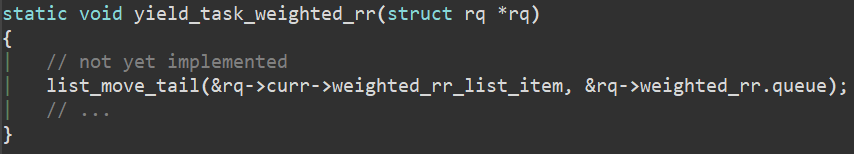
1. **dequeue\_task\_weighted\_rr()**

呼叫update\_curr\_weighted\_rr更新ready queue，並用list\_del把task從ready queue中移除，並把ready queue中的數量減一。程式碼如下圖所示：



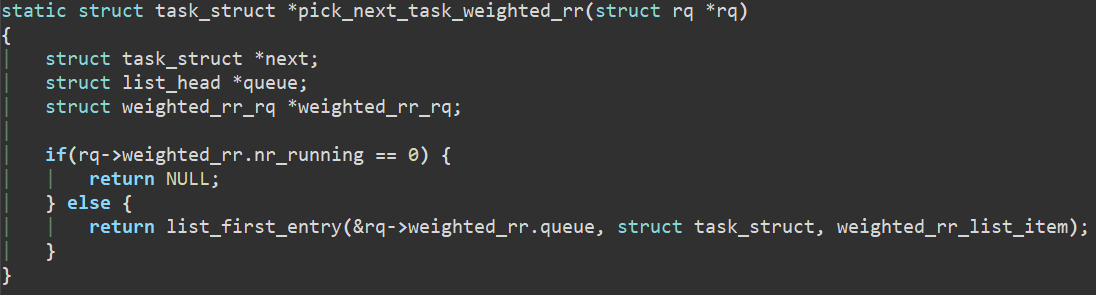
1. **yield\_task\_weighted\_rr()**

讓剛跑完的task把cpu資源讓出，且用list\_move\_tail將剛佔用cpu資源task移到ready queue的最尾端。程式碼如下圖所示：



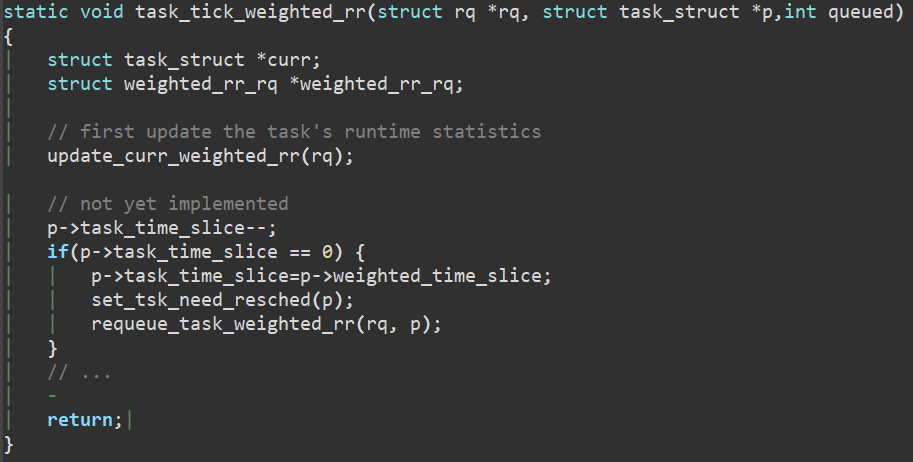
1. **pick\_next\_task\_weighted\_rr()**

若ready queue中沒有task則回傳NULL，若有則用list\_first\_entry取出ready queue中最前面的task。程式碼如下圖所示：



1. **task\_tick\_weighted\_rr()**

更新現在的ready queue，將task的time slice減一，最重要的是若task 的time slice為0，則更新time slice，並呼叫requeue\_task\_weighted\_rr把task排到ready queue的最後。程式碼如下圖所示：



* **Result**

在Linux裡面實際執行結果如下：

