**Operating System Homework 3 Report**

Student ID: 0410137

Name: 劉家麟

**Detailed description of the implementation:**(Number of threads, the purpose of those threads, how do you use mutex lock and semaphore…etc.)

Hw3-1: Blur #threads = 8

|  |
| --- |
| Lock0[3] |
| Lock1[3] |
| Lock2[3] |
| Lock3[3] |

在第一題，我用的是**mutex**。

我把圖片拆成4塊，並用鎖來控制RGB2Grey跟Gaussian Filter的先後順序。

一開始直接先宣告4個thread來做RGB2Grey的部分。

後來再宣告另外4個thread來做Convolution，但因為Convolution用的Filter會影響到前後的區塊，所以我用鎖來控制convolution的開始。

每個區塊的開始須先滿足「前面那塊做完RGB、自己做完RGB、後面那塊開始做了部分RGB」才能做Convolution，lockx[0]&lockx[1]代表做完了，lock[2]則代表已經做了部分。

Hw3-2: Sobel #threads = 8

|  |
| --- |
| Lock0[2] |
| Lock1[2] |
| Lock2[2] |
| Lock3[2] |

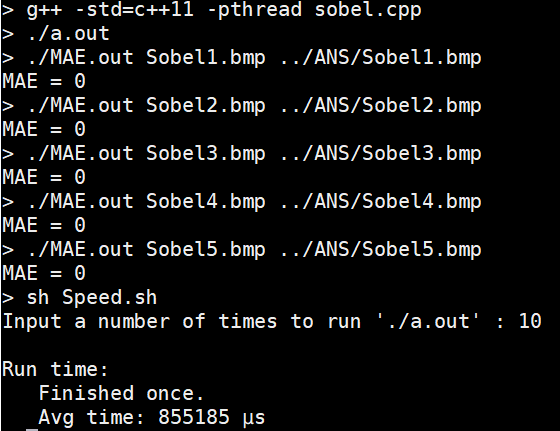
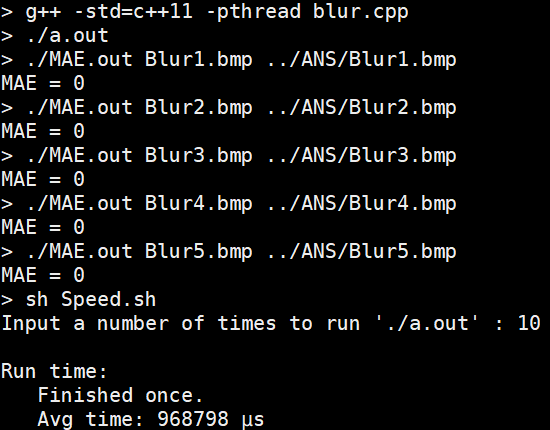
在第一題，我用的是**semaphore**。

第二題其實是直接延續第一題的概念，一樣把圖片拆成4塊分別做RGB2Grey以及Convolution(共8個thread)，再用鎖來控制RGB2Grey跟Gaussian Filter的先後順序。

但因為第二題用semaphore，所以就沒有宣告那麼多鎖，做完的部分一次可以允許2個thread進入，所以在一開始鎖2次後，做完直接一次釋放2個鎖，讓convolution在要求時，可以兩個thread都要求同一個做完的lock。

**Your speed:**

每次執行結果都不太一樣，兩個Speedup大概都1.5~2.0之間。



**Problems encountered and solutions:**

1. 一開始做sync的部分一直有問題，常常圖片時好時壞，本來一直以為是硬體thread不夠的問題，後來想一想，可能是要求鎖的部分沒有鎖好。  
   本來鎖在進RGB2Grey各自的thread之後才鎖，後來把鎖在宣告thread前就先鎖好，進入thread做完RGB2Grey才釋放後，output就沒有問題了。
2. 一開始是先做好部分的RGB2Grey在開始做convolution，結果加速超慢，只有1.03倍，怎麼樣都無法提升。後來就決定拆圖片再加上同時做，不過同時做還是會有sync的問題，有時候有的區塊沒做好。最後把鎖放對位置整體加速終於提升到1.5倍以上。