**Operating System Homework 3 Report**

Student ID: 0416208

Name: 黃士軒

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

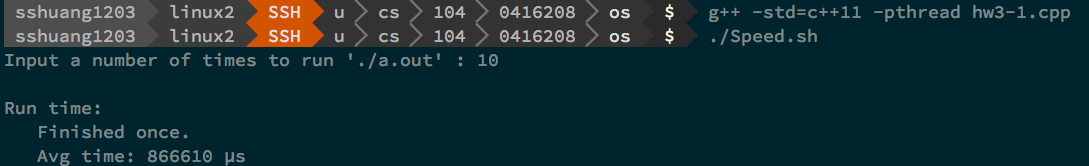
HW 3-1: 一共用了6個Thread，主要分為三個階段，thread\_rgb2grey, thread\_gaussian, thread\_extend\_size，這三個階段分別有兩個thread分工運行。同時，在做thread\_gaussian時，利用sem\_post和sem\_wait來讓thread\_gaussian, thread\_extend\_size協調運作，以達成multithread的效果。

HW 3-2: 一共用了10個Thread，主要分為四個階段，thread\_rgb2grey, thread\_edgefilter, thread\_compute, thread\_extend\_size，這四個階段，除了edgefilter有分GX GY所以有4個thread，其他分別有兩個thread分工運行。同時，在做thread\_compute時，利用pthread\_mutex\_lock

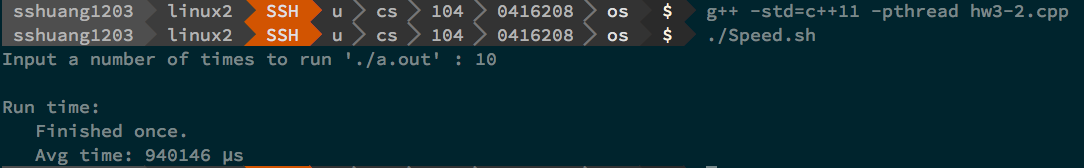
和pthread\_cond\_signal來讓thread\_compute, thread\_extend\_size共同協調分工運作，以達成multithread的效果。

**Your speed:**

HW 3-1 -> 1.78倍



HW 3-2 -> 1.52倍



**Problems encountered and solutions:**

在做multithread的時候，真的很容易遇到資料不同步的問題，但是同時要進行平行運算降低mutex或是semaphore對速度的影響真的不容易，加上server只有兩個processors，所以其實一次開太多thread反而造成overhead，在經過眾多嘗試發現4的thread是最佳化的效率的最高thread數量。