**Operating System Homework 4 Report**

Student ID: 0510002

Name: 袁鈺勛

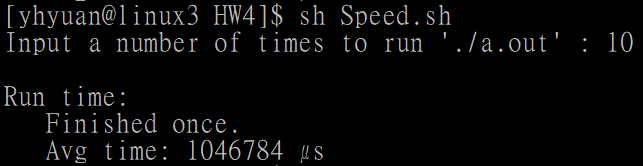
**Detailed description of the implementation:**(Number of threads, the purpose of those threads, how do you use mutex lock and semaphore…etc.)

　　在第一題裡面我用了兩個thread，分別是做mean filter和sobel filter，然後在創建mean filter之前，會先讓main function拿到lock並且mean filter在創建後會排隊拿lock，再由main function以condition wait的方式釋出lock給mean filter，以此方式確保mean filter會先跑。等到mean filter跑到第五個row的時候便會signal main function，main function才會從wait中啟動並創建sobel filter。

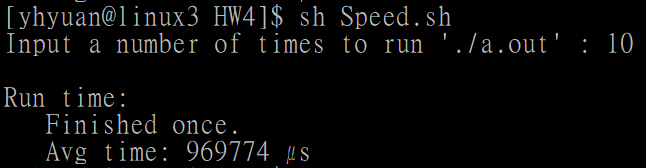
　　在第二題裡面我只比第一題多加了一個RGB2grey的thread，而且將原先mean filter優先啟動的機制移到了RGB2grey，同樣的RGB2grey和mean filter會在run到第五個row時讓main function啟動下一個thread

**Your speed:**

第一題出現過的最快時間是1.047秒



第二題出現過的最快時間是0.970秒



**Problems encountered and solutions:**

　　這次的作業因為要考慮到thread的優先順序問題，而且之前的作業也沒有用到lock，所以在做作業的時候常常會dead lock。一開始遇到thread順序的問題時是在child裡面加入一些output來看狀況才發現thread不會依照create的順序執行，所以用mutex lock來保證順序，但加入lock後有時會忘記要unlock，都要真的執行code後才發現。