109590004 呂育瑋 作業系統 HW#3

系統環境：VMware, Linux, Ubuntu 18.04

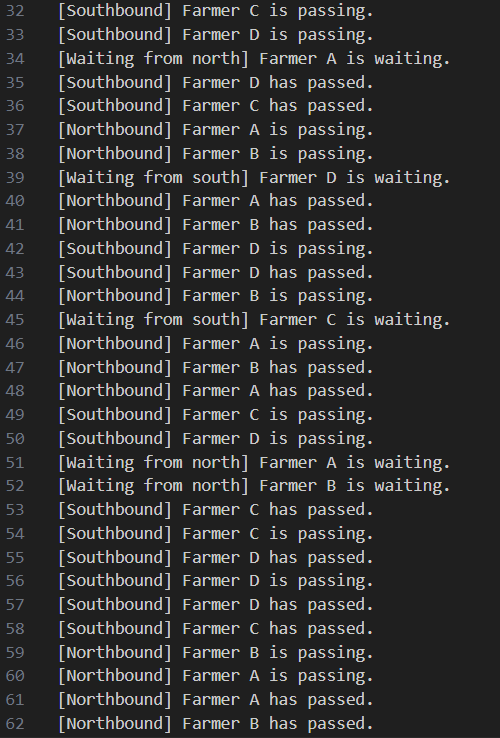
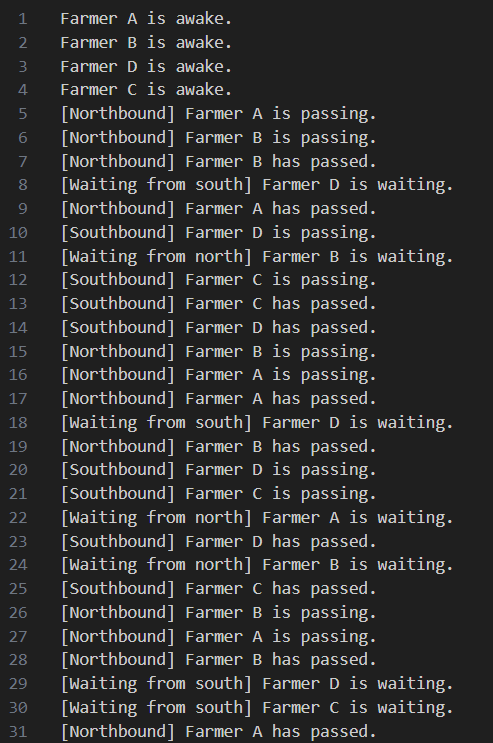
* Problem 7.17

創建兩種進程函式（北方與南方），使用mutex lock保護共享變數，以及condition variable判斷是否等待，因為要避免雙向的農夫在橋上碰面，而同方向的農夫同時在橋上是允許的，所以等待的判斷式為：只有當沒有任何對向農夫在橋上才可以過橋，否則需等待直到所有對向的農夫經過橋，農夫過橋後發送廣播給所有對向的農夫，通知他們可以不需要等待，但如果對向農夫判斷還有農夫往他們這個方向過橋，則繼續等待，周而復始。

程式碼：Source code\Chap7.17\farmers.c

檔案編譯方式：gcc -o farmers farmers.c -lpthread

輸出：Source code\Chap7.17\farmers\_output.txt



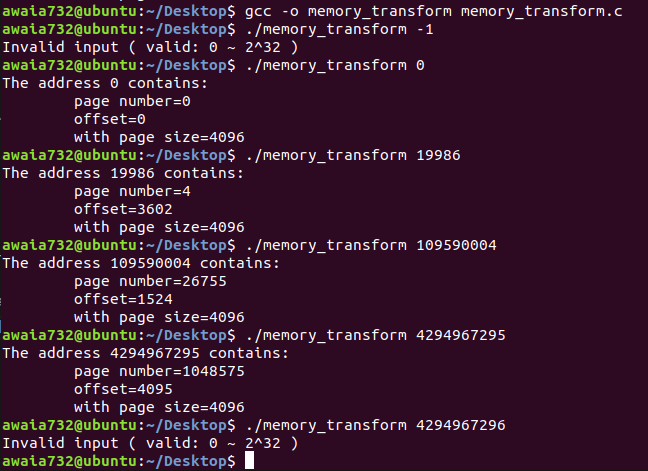
* Problem 8.25

輸入一個32-bits的十進位數字，設輸入的資料型別為unsigned int，確定執行程式有輸入參數，且用unsigned long來判斷輸入的數字範圍是否在32-bits中，設輸入為virtual address，若page size = 4086，則page number = virtual address / page size（取商）、offset = virtual % page size（取餘）。

程式碼：Source code\Chap8.25\memory\_transform.c

檔案編譯方式：gcc -o memory\_transform memory\_transform.c

輸出： Source code\Chap8.25\memory\_transform\_output.png



* Problem 9.26

先創建代表頁面參考的隨機字串，並依照以下三種方法進行頁面替換

FIFO：用一個Queue來決定替換的順序，使用一個陣列當作queue，並用queue index來當作queue對列的開頭，以此來決定哪一個頁面應該被替換掉。

Optimal：尋找在frame中最久才會被再次呼叫的頁面做替換，這能得到最少的page fault，但在實際應用時很難實現。

Less-Recently-Used：尋找在frame中閒置最久的資源做替換，是最常被實際應用的方法。

程式碼：Source code\Chap9.26\page\_replace.c

檔案編譯方式：gcc -o page\_replace page\_replace.c

輸出：Source code\Chap9.26\page\_replace\_output.png

