

### 作業三：檔案鎖

作業目的：

- 了解在作業系統中，如何讓多個程式同時操作一個檔案
- 了解 flock 及 lockf 的 overhead 究竟是多少

題目：請撰寫程式。

第一支程式稱之為 **flock**，他使用一個迴圈，共執行 1000 次，在迴圈內先等待 0.1 秒，然後在迴圈中將 flock.db 的最後一個數字讀出來(假設為 X)，並使用 lseek 將檔案指標往前移動 X 個 byte，然後寫入值「X+1」。這個程式使用 flock 函數確保一次只有一個程式更新 flock.db。flock.db 的第一個值是 3500。

第二支程式稱之為 **lockf**，他使用一個迴圈，共執行 1000 次，在迴圈內先等待 0.1 秒，然後在迴圈中將 lockf.db 中的最後一個數字讀出來(假設為 X)，再使用 lseek 將檔案指標往前移動 X 個 byte，然後寫入值「X+1」。這個程式使用 lockf 函數確保一次只有一個程式更新 lockf.db。lockf.db 的第一個值是 3500。

測試：

1. 如果同時執行 4 個 lockf (/flock)，這些程式可以同時執行，並更新 lockf.db (flock.db) 嗎？請問總共花地多少時間？
2. 如果多個 lockf (/flock) 在更新 lockf.db (/flock.db) 的時候，你使用 vim 或其他編輯器打開 lockf.db (/flock.db)，請問此時會發生什麼情況？如果儲存的話，會發生什麼樣的情況？
3. 請問你的檔案的實際大小與邏輯大小？你有辦法『大致上』推算出檔案大小。

提示：lockf、flock、nanosleep、sprintf、ls -lh、time、“echo >”

繳交：

1. 程式碼和 makefile，助教執行 make 指令時，必須自動產生 lockf 和 flock 二個執行檔案，並且產生 lockf.db 和 flock.db。
2. 測試報告，測試報告的格式並須為 pdf。測試報告前請附上姓名（可隱匿一個字）及學號
3. 請將所有檔案壓縮成.tar.bz2。繳交到 ecourse2 上
4. 不能遲交
5. 再次提醒，助教會將所有人的作業於 dropbox 上公開

6. 如果真的不會寫，記得去請教朋友。在你的報告上寫你請教了誰即可。