

#### 作業四：buffer 與效能

##### 學習目標：

1. 了解 linebuffered、unbuffered、fullybuffered 的意義
2. 了解在 fullybuffered 的情況下，buffer 超過某個大小效能就不會再增加
3. Unbuffered 的 libc 的 function call 及 system call 的對應幾乎是一對一
4. 了解 ungetc 與 EOF 的關係（選項）

##### 附註：

1K = 1024

1M = 1024 X 1K

題目：量測 buffer size 對效能的影響。

寫一支程式，程式的名稱是 fileperf，為這支程式接受二個參數，第一個參數是 source\_file，第二個參數是 dest\_file，第三個參數則是設定 buffer\_size，是一個數字，值可能是-1、0、或大於 0 的任何正整數（不會超過 20MB）。

```
%~ ./fileperf source_file dest_file_1 1000 /*buffer size 為 1000 byte*/
```

```
%~ ./fileperf source_file dest_file_2 0 /*unbuffered, 0, 1024*/
```

```
%~ ./fileperf source_file dest_file_3 -1 /*linebuffered, 1024*/
```

source\_file 是一個幾乎是「英文」文字檔案（如果遇到並非 ASCII 的字元可以忽略，或者處理這些字元）。將 source file 的內容複製到 dest\_file。每一行的字數不能超過 80 個英文字（如果遇到要雙位元顯示的，例如：中文字，那就當成一個或二個字都可以）。

請參考 hw2\_example 目錄裡面的範例。記得，每一行最前面和最後面的空白要去掉，只有英文單字間的空白才可以換行。此外不會有單字是超過 80 個字。Buffer 的設定同時套用到 dest\_file 和 source\_file。

##### 測試：

1. 請問 buffersize 分別是：0、-1、4KB、16KB、64KB、1MB、8MB 的執行速度分別為何？（使用 time 指令）
2. 使用 ltrace 觀察你的應用程式呼叫「函數庫的情況」
  - 甲、對，ltrace 我沒教過
  - 乙、嗯，也沒人教過我啊，我是自己上網學的
  - 丙、會用 ltrace，看到別人寫出很厲害的程式，可以偷看一下他呼叫了什麼函數庫。
  - 丁、使用方法跟 strace 很像

- 戊、call (x86, asm)
3. 使用 strace 觀察你的應用程式呼叫「作業系統的情況」  
甲、syscall (x86, asm, trap, software interrupt, 大概是 call(x86, asm)的速度的 1/500)
  4. 有辦法根據 2 和 3 分析一下「呼叫作業系統核心函數 (system call)」和「函數庫呼叫」的「成本」差異嗎？  
甲、clock\_gettime(), 在 x86 與 ARM 上的理論精準度為 1 nanosecond  
乙、clock\_gettime(&t1); f(); clock\_gettime(&t2); return t2-t1;

提示：

1. 功能類似 `fold -w 80 -s text.txt`
2. 或許會用到 unget() 函數，如果真的用到 **要注意 unget() 和 EOF 的關係**
3. **如果要將結果寫出，『幾乎我們都要先把檔案長度變為 0』**，假設原本的檔案大小是 41,387 byte，而我們寫入的資料只有 2,011 byte，那麼在 2,012 byte 以後的資料會是「舊檔案的資料」。這應該不是我們想要的（除非我們真的要局部更新檔案）。
4. **Unicode 字串** 可以使用 <uchar.h> 處理
5. 這個程式的功能和 `$ fold -w 80 -s input.txt > output.txt` **幾乎** 一模一樣

#### 程式碼的評分重點：

1. 能使用 setvbuf 設定「讀取」和「寫出方式」
2. 將 input file 格式化成 80 字一行，不可以在英文字中間換行
3. 在上述前提下，可以將空白留在上一行或者下一行或者忽略
4. 最好能處理 wchar（自我修煉，助教不會特別去看是否處理了這部分）

繳交：

1. 程式碼和 makefile，助教執行 make 指令時，必須自動產生 fileperf。
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. **繳交期限：如課程網頁上的公布**
7. **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。**