

# 作業三

## ● 作業內容

### 第一部分

- 設計一個實驗，了解作業系統配置記憶體的方法
- 經由這個實驗回答：malloc 40MB 時，作業系統是否會立刻配置 40MB 的記憶體給該程式
- 了解作業系統是需求分頁，當你跟作業系統要求配置某個數量的記憶體，實際上作業系統不會馬上給你，除非真的使用該快記憶體，例如對記憶體內每一個 byte 進行讀寫

### 第二部分

- 撰寫一支程式可要求大量的 memory，迫使 system 做 swap，之後使用 free() 使 used memory 變成 free memory，由於曾經 swap 過因此最後的 free memory 會比之前多。
- 了解作業系統是以全域的方式配置記憶體，當跟系統要很多記憶體，就會將其他 process 的記憶體『擠出去』到 swap space
- 可輸入一個參數 int，表示要求多少 GB 的 memory。例如：  
./free\_mem 2，表示 2\*1024\*1024\*1024 byte。
- 必須提供觀察方式，但不可使用 GUI、系統監控（相當於 Windows 的系統管理員）。

## ● 作業要求

1. 第一部分的執行檔名稱要為 “my\_malloc”
2. 第二部分的執行檔名稱要為 “free\_mem”
3. You are responsible for writing a **makefile** suitable for your whole program
4. The document of your program will be named “**readme.pdf**”
5. Your program should must be run successfully on your owned **Ubuntu**, workstation is not allowed.

Readme (40%) =

- 第一個實驗
  - 說明你實驗的依據，
  - 說明你所設計的程式是否有特別之處
  - 你觀察 Linux 中那些檔案或者資訊，以驗證你的論述
- 第二個實驗
  - 說明你的程式花費了多少時間，釋放了多少記憶體
  - 說明你的 free\_mem 對於 swap space 的影響（即：是否額外造成 page-out 及 page-in），如何觀察？

MakeFile (10%)

Source Code (50%)

- **繳交期限**

**2017/12/4 23:59:59 前 不可遲交, 也不接受任何理由補交**

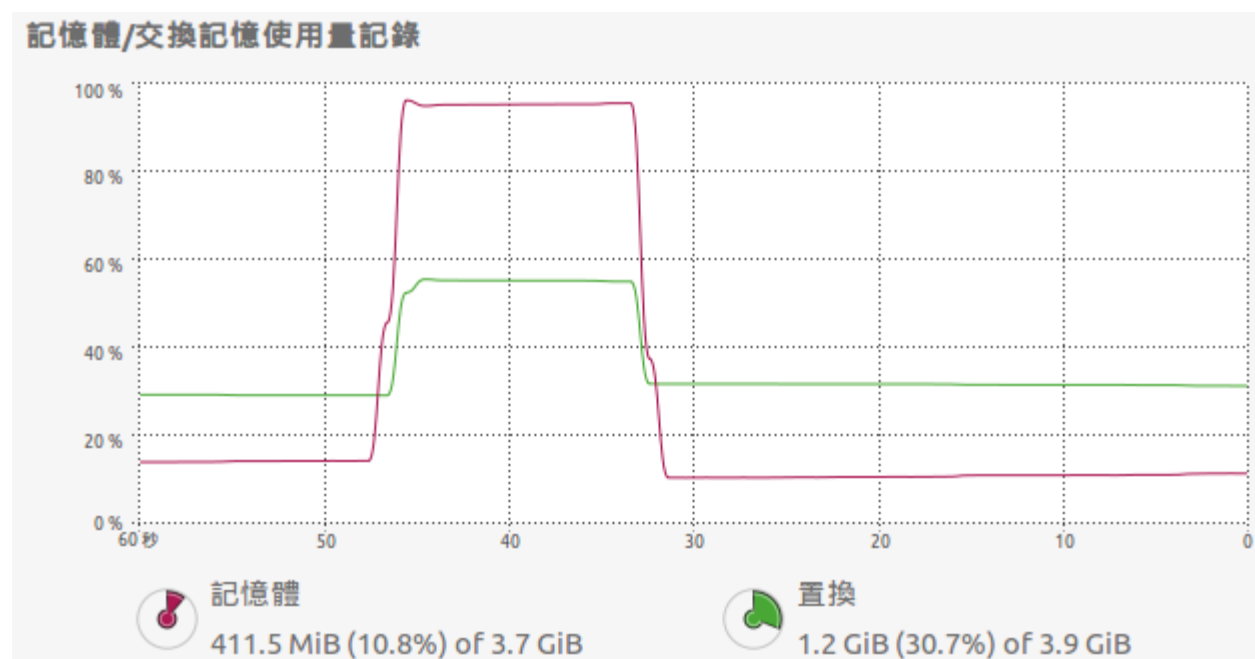
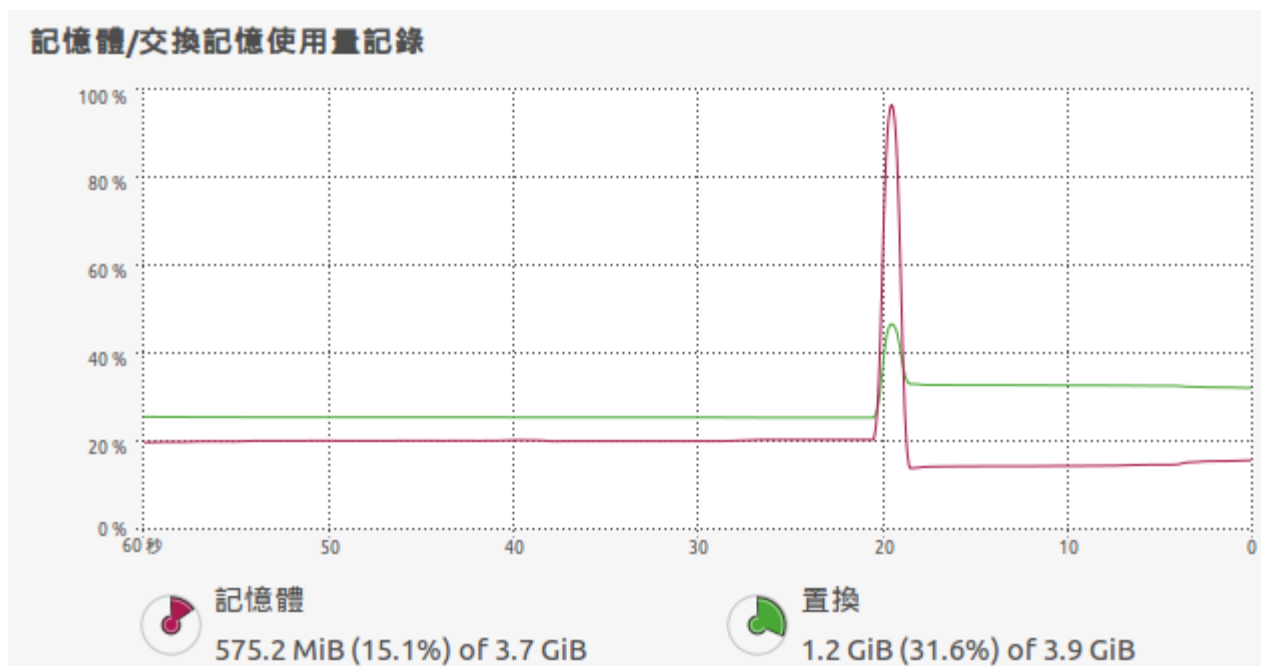
- **繳交方式**

將你的所有檔案壓縮成 **學號\_hw3.zip 格式** 後上傳到 E-course 上的 “**作業 3**”

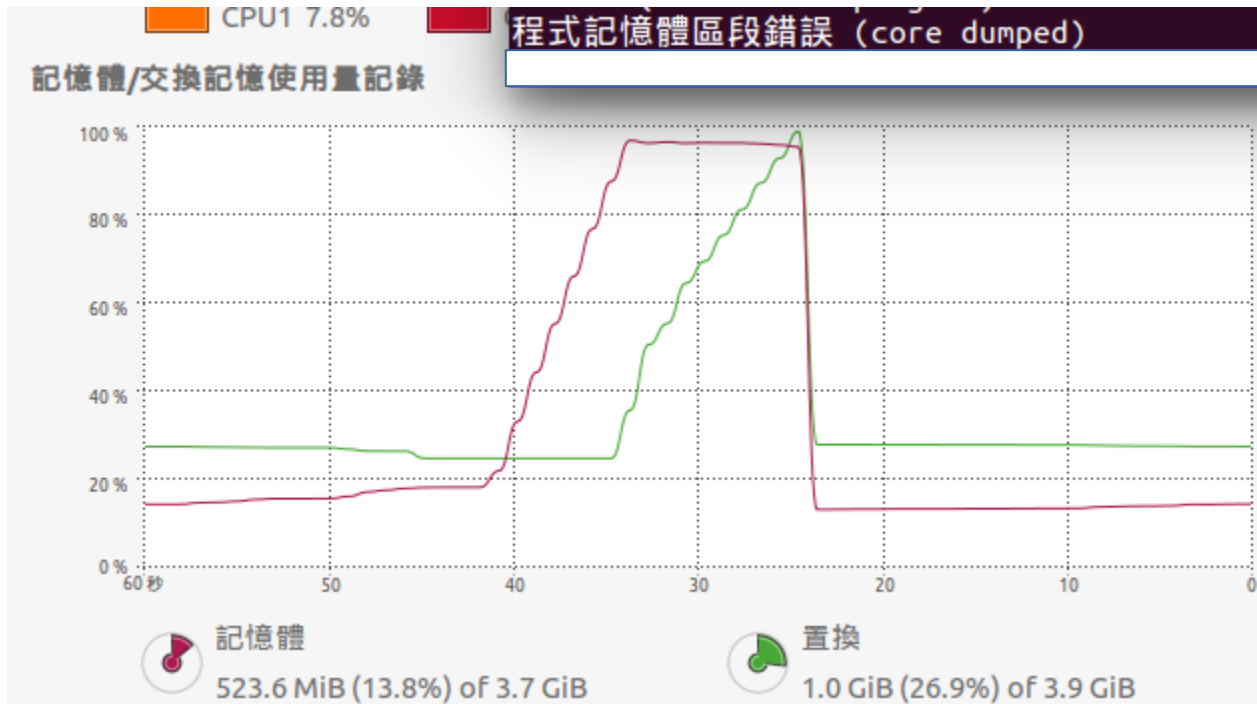
還有第 3 頁補充說明、注意事項

## 第二部份注意事項

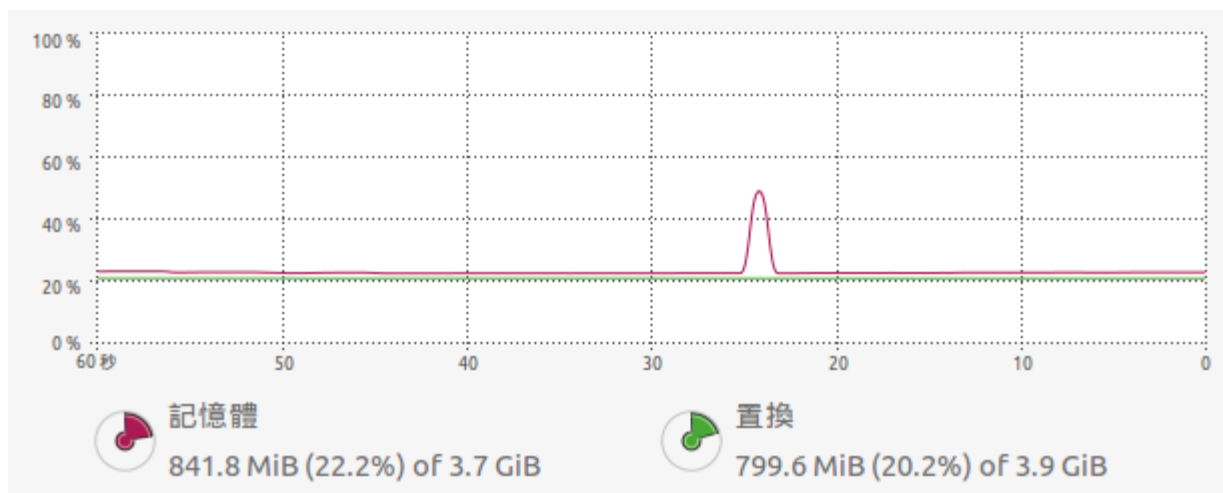
- 理想狀況：發生 swap、有 free()，有無暫停皆可



- 扣分狀況：灌爆 mem, swap、被 system 強制中斷、loop、不正常關閉、完全當機



- 輸入參數的意義：以前有些程式在 VM 中很理想，但在改作業的時候：



要求 1GB，但是在實機無法觸發 swap。自訂大小確保能適用於任何環境。

- 以上只是方便說明，作業禁止使用 GUI