OS第二次作業

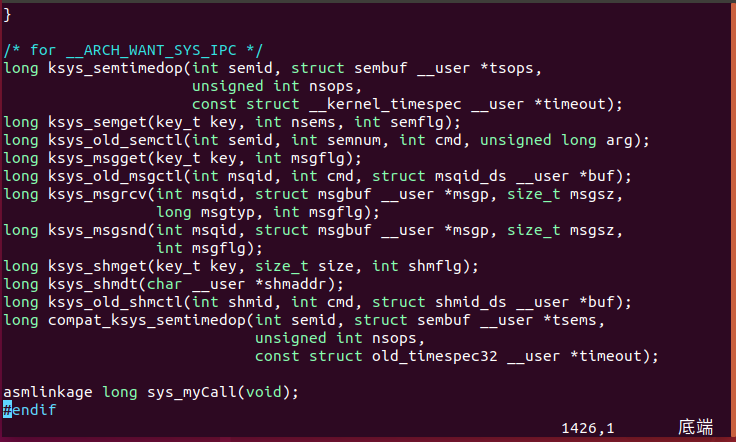
一、分工

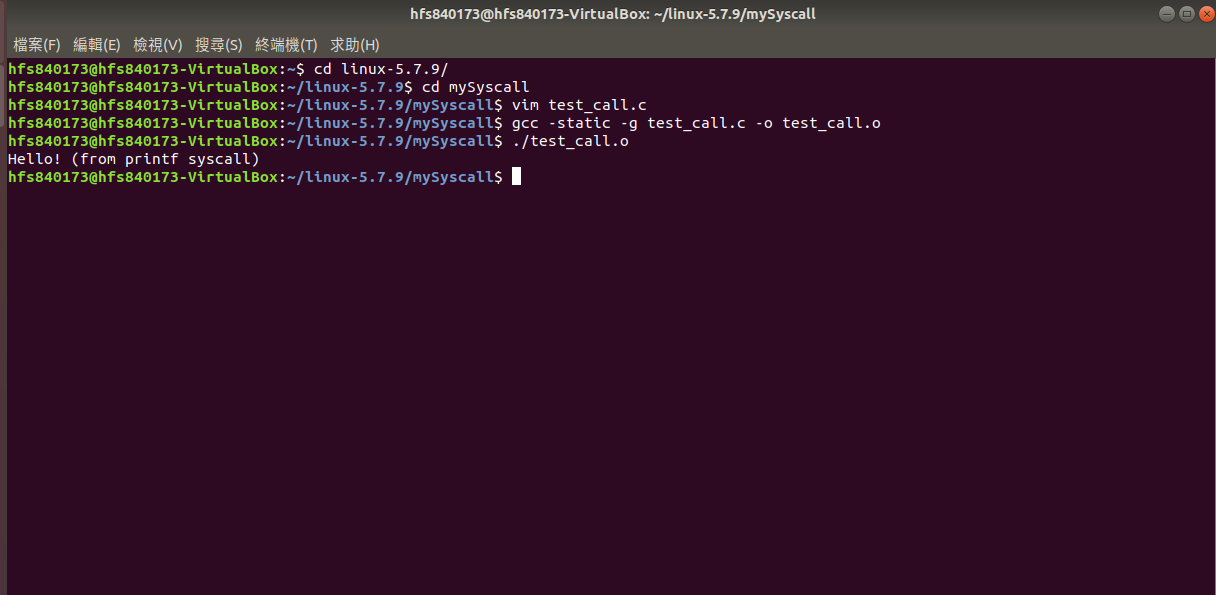
鄧智宇：撰寫報告、相關資料查找

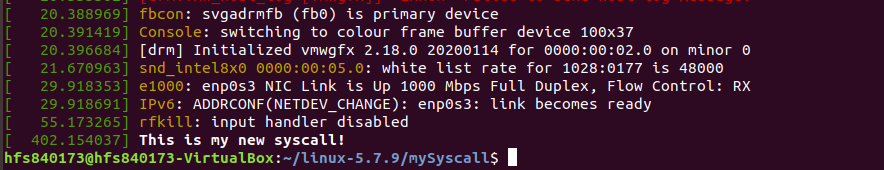
黃信維：撰寫程式、相關資料查找

二、作業步驟以及遇到問題

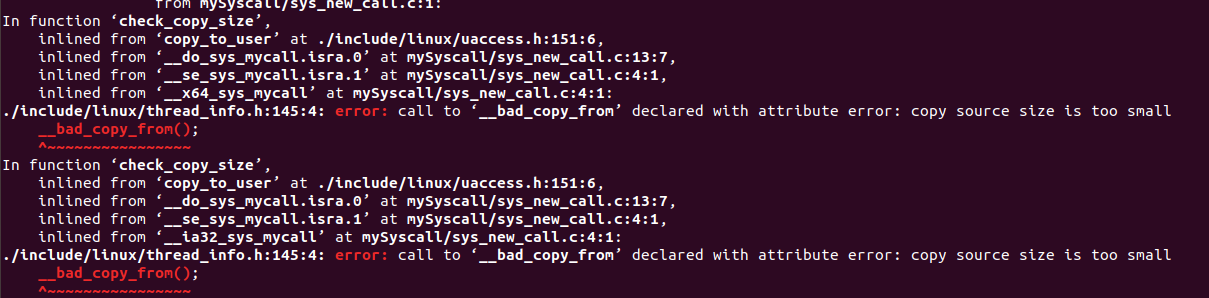
1. 一開始執行基本的步驟，編輯、新增新的system cal等，還有重燒作業系統

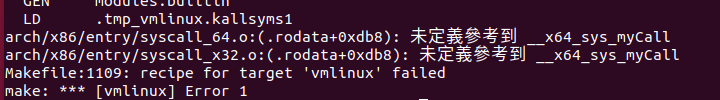






測試程式可成功執行。

1. 遇到的第一個問題是每次燒的速度太慢，經過助教提醒，可將虛擬機的核心樹調高，並使用make –j2，就快多了。
2. 在撰寫syscall的時候，一開始想用直接使用組合語言的方式，但是發現印不出任何東西。後來經提醒，去查了SYSTEM\_DEFINEx的用法，並在syscalls.h內找到相關定義。
3. 因為我們要傳入四個參數，所以是用SYSTEM\_DEFINE4，然後把加減號、和兩個整數和答案分別傳入自定義的system call裡，並在system call內運算。運算完再傳回user space後印出。
4. 另外我們也查詢了copy­\_to\_user的使用方式，以把內容從kernel space複製到user space。
5. 在編譯時出現下面的訊息：解決方式是將copy\_to\_user的參數改為sizeof()，就可以成功編譯了
6. 但是在make 的時候，卻跑出錯誤訊息如下

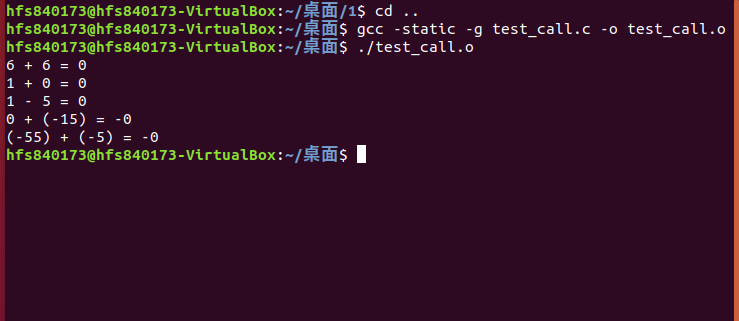


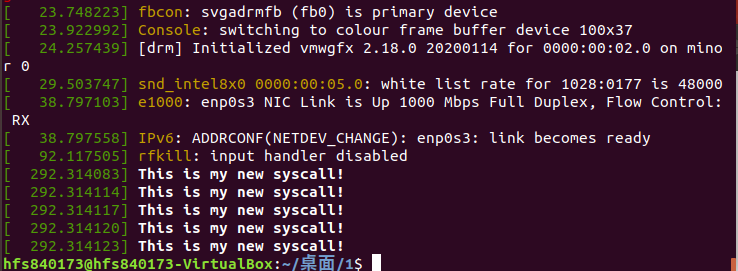
查到的解決方法是，linux的版本是新的，但C語言函式庫沒有同步更新所導致，所以我們更改syscall\_64.tbl重新make一次，但因為時間來不及燒完，所以先交沒有make成功的。

三、浮點數是否能在 syscall 內使用？

大部分的觀點都是認為盡量不要，原因可能有浮點數的暫存器較耗空間和資源等，但仍可以使用。對於有FPU的處理器，因為linux會將浮點數轉成整數後運算，所以要讓module使用硬浮點，也就是先改kernel的配置(Makefile內)，然後再行編譯；然後就可以執行了。但如果沒有FPU，就只能用軟浮點來模擬期運算。

四、部份成功截圖：





五、引用資料

查copy\_to\_user、copy\_from\_user的用法

<https://stackoverflow.com/questions/6515227/source-code-example-from-linux-kernel-programming>

<https://stackoverflow.com/questions/29397364/copy-to-user-and-copy-from-user-for-basic-data-type>

<https://elixir.bootlin.com/linux/v5.7.9/source/include/linux/uaccess.h#L149>

為了要查詢系統SYSCALL\_DEFINE的定義

<http://gityuan.com/2016/05/21/syscall/>

<https://elixir.bootlin.com/linux/v5.7.9/source/include/linux/uaccess.h#L149>

能否使用浮點數

<https://stackoverflow.com/questions/15883947/why-am-i-able-to-perform-floating-point-operations-inside-a-linux-kernel-module/47056242>

<https://stackoverflow.com/questions/13886338/use-of-floating-point-in-the-linux-kernel>

<https://blog.csdn.net/k7arm/article/details/81406842>

<https://www.itread01.com/p/1389126.html>

編譯時參數問題的解決方式

<https://github.com/analogdevicesinc/linux/pull/206>

<https://blog.csdn.net/u013969018/article/details/86686842>

「未定義參考」解決方式

<https://stackoverflow.com/questions/29137244/undefined-reference-error-while-using-custom-system-call>