**簡易shell作業  
1094841 李昱佑**

**繳交文件說明：**

my\_shell.c為我所撰寫的程式碼，my\_shell為my\_shell的執行檔。

而child.c、child、c3、c3.c、guess\_ab、guess\_ab.c為老師所提供的子程式。

**功能規劃說明:（紅字為新加入的功能）**

1. my\_shell一開啟會先輸出自己的pid，如” Parent(my\_shell) pid is xxx”。
2. 無論是否為老師提供的程式，只要在my\_shell輸入其他程式的路徑(如：linux內建的/bin/ls)，皆能正常的運行使用者所指定的程式。
3. 只要在my\_shell提示符號後的輸入中最後為&，my\_shell就不會等待child；反之，輸入的最後無&，則my\_shell就會等待child執行完畢。
4. 當child執行結束後，回到my\_shell時，my\_shell會輸出剛結束的child的pid與status()，如” I'm parent(my\_shell). Ended child pid: xxx, status: xxx”。
5. 如果輸入的child program不在目錄中或輸入的絕對路徑有誤，my\_shell會output “xxx is not found”。
6. 如果execl執行失敗，會將fork出的child結束掉。
7. 如果fork()失敗，會輸出fork()失敗的資訊。

**my\_shell程式說明: (my\_shell.c裡已寫入完整註解)  
一張含有 文字 的圖片

自動產生的描述**

**一張含有 文字 的圖片

自動產生的描述一張含有 文字 的圖片

自動產生的描述一張含有 文字 的圖片

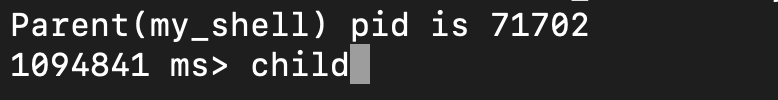
自動產生的描述一張含有 文字 的圖片

自動產生的描述**

**操作說明:**

與老師demo的三部影片中的操作方式相同：

1. 在原本的shell上寫 ./my\_shell 就能開啟my\_shell
2. 在my\_shell提示符號旁可以打入相對路徑或絕對路徑來執行指定的程式

 或 一張含有 文字 的圖片

自動產生的描述

1. 若輸入的最後有&，則會my\_shell不會等待child，兩程式會同時執行。
2. 若輸入最後沒有&，則my\_shell會等待child執行完後再執行。
3. 除了上述，也能執行任何其他程式，如下圖：

一張含有 文字 的圖片

自動產生的描述

1. 當child執行完回到my\_shell，my\_shell會顯示已執行完的child的pid與exited status，如上圖。
2. 若輸入的程式不存在於目前目錄或不存在與電腦中，則會顯示下圖:

一張含有 文字 的圖片

自動產生的描述

1. 其他細節與老師的demo影片中一樣。