

Assignment6

- Code

```
1  #include <stdio.h>
2  #include <stdlib.h>
3  #include <unistd.h>
4
5  int main() {
6      system("ps -a");
7      int pid;
8      if ((pid = fork()) < 0)
9          perror("fork error");
10     else if (pid == 0)
11         return 0;
12     system("ps -a");
13     return 0;
14 }
15
```

- Line 6

印出進入 main function 一開始的 process status.

- Line 8 to Line 11

parent 利用 fork() 呼叫一個 child. 如果目前在執行的是 child 就直接 return 0, 並且不設定任何 wait 或 waitpid 讓他的 parent 知道自己已經 return. 如此一來 parent 不會釋放 child 的記憶體空間。

- Line 12

印出目前的 process status. 可以看到與 Line 6 不同的是多了一列 STAT 顯示 Z+, 而 COMMAND 顯示 <defunct> 的 process. 這個就是我們刻意建立出的 zombie process.