

# HW13 pipe

408410019 資工二 徐怡娟

1. 已參考 pipe4-3.c 完成輸入 ./fileCount /YYY 會執行 ls -R /YYY | wc -l 的功能

正常執行完的結果

```
accepted@ubuntu:~/system-programming/ch11/HW13$ sudo ./fileCount /
first child group: 4614
first child group: 4614
second child group: 4614
ls: cannot open directory '/run/user/1000/doc': Permission denied
ls: cannot open directory '/run/user/1000/gvfs': Permission denied
1016554
```

2. 註冊 signal(SIGINT, handler); 後按下 ctrl-c 讓整個程式停止執行

將 ls(first child)設置為新的 process group leader，再把 wc(second child)加入

為 first child process group 的成員程序，按下 ctrl-c 後 OS 會發送 SIGINT 給前

景運程序(parent)，執行 handler 的動作：kill(-1\*pid1, signum); 發送給 first

child process group 中所有成員 SIGINT 的訊號，因此所有程序都會被中止。

```
/*parent註冊signal handler*/
signal(SIGINT, signal_ctr_c);

void signal_ctr_c(int signum) {
    fprintf(stderr, "kill process group %d\n", -1*pid1);
    //殺掉process group
    kill(-1*pid1, signum);
    //parent結束離開
    _exit(0);
}
```

```
accepted@ubuntu:~/system-programming/ch11/HW13$ sudo ./fileCount /
parent group: 4967
first child group: 4969
second child group: 4969
ls: cannot open directory '/run/user/1000/doc': Permission denied
ls: cannot open directory '/run/user/1000/gvfs': Permission denied
^Ckill process group -4969
```

3. 未註冊 `signal(SIGINT, handler);` 後按下 `ctrl-c` 讓整個程式停止執行

`ls`(first child)和 `wc` (second child)都是由同一個 parent fork 出來的，會繼承

parent 的 process group id(等同 parent pid)，因此 parent 和兩個 child 屬於同

一個 process group 中，按下 `ctrl-c` 後 OS 會發送 `SIGINT` 給前景運行政序

(parent)，執行預設的動作：“中止”同 process group 的所有程序。

```
/*parent註冊signal handler*/  
//signal(SIGINT, signal_ctr_c);
```

```
accepted@ubuntu:~/system-programming/ch11/HW13$ sudo ./fileCount /  
parent group: 4934  
first child group: 4934  
second child group: 4934  
ls: cannot open directory '/run/user/1000/doc': Permission denied  
ls: cannot open directory '/run/user/1000/gvfs': Permission denied  
^C
```