

HW4 - Readme

1. 在 Linux 環境下進行編譯 p1.c、p2.c、p3.c

用下面指令進行編譯

```
gcc -o p1 p1.c sem.c
```

```
gcc -o p2 p2.c sem.c
```

```
gcc -o p3 p3.c sem.c
```

如果出現以下情況，可以直接忽略 → 不影響執行

```
sem.c: In function 'create_sem':
sem.c:21:13: warning: implicit declaration of function 'ftok' [-Wimplicit-function-declaration]
   21 |     ipc_key = ftok(path,name);
       |               ^~~~~
sem.c:22:11: warning: implicit declaration of function 'semget' [-Wimplicit-function-declaration]
   22 |     semid = semget(ipc_key, 1, IPC_CREAT| 0666) ;
       |             ^~~~~~
sem.c:28:11: warning: implicit declaration of function 'semctl' [-Wimplicit-function-declaration]
   28 |     ret = semctl(semid, 0, SETALL, arg);
       |            ^~~~~~
sem.c: In function 'P':
sem.c:53:5: warning: implicit declaration of function 'semop' [-Wimplicit-function-declaration]
   53 |     semop(semid, &decrease, 1 );
```

2. 編譯完成後，依序執行 p1、p2、p3

用下面指令依序執行 p1、p2、p3

```
./p1 &
```

```
./p2 &
```

```
./p3 &
```

3. 如果想更換執行順序為 p1、p3、p2，先清除現存的 semaphore

先使用以下指令查看目前的 semaphore id (**semid**)

```
ipcs
```

再使用以下指令清除所有 semaphore，{semid} 換成你看到的 semid

```
ipcrm sem {semid}
```

4. 依序執行 p1、p3、p2

```
./p1 &
```

```
./p3 &
```

```
./p2 &
```

輸出結果應該與步驟2的依序執行 p1、p2、p3 相同