

## Ch05.02 OS 作業

### 問題一

1. 請修改「鎖定」和「解鎖」相關的程式碼，使得「上鎖時」可以紀錄到底是誰拿到這個鎖（修改 signal handler，每一秒鐘印出「如果鎖定了，那麼是誰上鎖」）

(1) void simple\_spinlock\_lock(atomic\_int\* lock, int name)

- 新增傳入 name(行員編號)並且有 name 鎖住就存入 lock\_ary 紀錄

```
void simple_spinlock_lock(atomic_int* lock, int name) {
    int isunlock=0;
    int i;
    while (*lock == 0 && atomic_compare_exchange_weak(lock, &isunlock, name) == false) {
        lock_ary[i] = name;
        i++;
        isunlock=0;
    }
}
```

(2) void moneyTransfer(void \*givenName)

- 將 printf 的 name+1
- 傳入了 name，並且+1，因為要>0，若>0 才有辦法 lock

```
void moneyTransfer(void *givenName) {
    unsigned rand_value = (unsigned)getgid();
    int name = (intptr_t)givenName;

    int source; int dest; int amount;
    //KPI: 代表這個行員共處理多少次轉帳
    long KPI=0;
    printf("%d號行員上線\n", name+1);

    //開始上鎖
    simple_spinlock_lock(&(lock_ary[source]), name+1);
    simple_spinlock_lock(&(lock_ary[dest]), name+1);
}
```

(3) void sigAlarm(int signo)

- 新增了 for 迴圈用 tmp 把 lock\_ary 取出後進行判斷，當 tmp != 0 印出幾號行員拿到鎖

```
void sigAlarm(int signo) {
    static int time = 9;
    time++;
    if (time == 15)
        stop = 1;
    printf("時間: %2d點\n", time);
    int i = 0;
    int tmp = 0;
    for (i = 0; i < 100000000; i++) {
        tmp = lock_ary[i];
        if (tmp != 0)
            printf("%d號行員拿到鎖.\n", tmp);
    } //for
    alarm(1);
}
```

(4) 運行結果

2 個行員每一秒印出誰拿到鎖

```
客戶 1998 有 100000 元
客戶 1999 有 100000 元
所有的人的錢共有：200000000
1號行員上線
2號行員上線
時間：10點
時間：11點
1號行員拿到鎖。
1號行員拿到鎖。
時間：12點
2號行員拿到鎖。
1號行員拿到鎖。
2號行員拿到鎖。
1號行員拿到鎖。
時間：13點
2號行員拿到鎖。
時間：14點
1號行員拿到鎖。
```