# Lab12 Report

> 12011702 张镇涛

## Q1

`local_intr_save(intr_flag)`的作用是在进入执行临界区代码前禁用中断，这样是和`local_intr_restore(intr_flag)`一起保证了操作系统在执行临界区的时候不会被打断，也就实现了同步互斥。

## Q2

(1)

**能够避免死锁。**

可以通过`sem_init(&s[i], 1)`确保只有1个哲学家能够拿起筷子，从而将多个线程的执行退化为单线程情况，以避免死锁。

(2)

**Code:**

```
//---------------part2-----------------
void phi_test_sema(int i)
{

    if(state_sema[i]==HUNGRY&&state_sema[LEFT]!=EATING&&state_sema[RIGHT]!=EATING)
    {
        state_sema[i]=EATING;
        up(&s[i]);
    }
}

void phi_take_forks_sema(int i)
{
      down(&mutex);
      state_sema[i]=HUNGRY;
      phi_test_sema(i);
      up(&mutex);
      down(&s[i]);

}

void phi_put_forks_sema(int i)
{
    down(&mutex);
    state_sema[i]=THINKING;
    phi_test_sema(LEFT);
    phi_test_sema(RIGHT);
    up(&mutex);
}
//---------------end-----------------

```

**Result:**

```
Iter 1, No.3 philosopher_sema is eating
Iter 2, No.0 philosopher_sema is thinking
Iter 1, No.1 philosopher_sema is eating
Iter 2, No.1 philosopher_sema is thinking
Iter 2, No.0 philosopher_sema is eating
Iter 2, No.3 philosopher_sema is thinking
Iter 2, No.2 philosopher_sema is eating
Iter 3, No.2 philosopher_sema is thinking
Iter 2, No.3 philosopher_sema is eating
Iter 3, No.0 philosopher_sema is thinking
Iter 2, No.1 philosopher_sema is eating
Iter 3, No.1 philosopher_sema is thinking
Iter 3, No.0 philosopher_sema is eating
Iter 3, No.3 philosopher_sema is thinking
Iter 3, No.2 philosopher_sema is eating
Iter 4, No.2 philosopher_sema is thinking
Iter 3, No.3 philosopher_sema is eating
Iter 4, No.0 philosopher_sema is thinking
Iter 3, No.1 philosopher_sema is eating
Iter 4, No.1 philosopher_sema is thinking
Iter 4, No.0 philosopher_sema is eating
Iter 4, No.3 philosopher_sema is thinking
Iter 4, No.2 philosopher_sema is eating
No.2 philosopher_sema quit
Iter 4, No.3 philosopher_sema is eating
No.0 philosopher_sema quit
Iter 4, No.1 philosopher_sema is eating
No.1 philosopher_sema quit
No.3 philosopher_sema quit
Iter 1, No.4 philosopher_sema is eating
Iter 2, No.4 philosopher_sema is thinking
Iter 2, No.4 philosopher_sema is eating
Iter 3, No.4 philosopher_sema is thinking
Iter 3, No.4 philosopher_sema is eating
Iter 4, No.4 philosopher_sema is thinking
Iter 4, No.4 philosopher_sema is eating
No.4 philosopher_sema quit
all user-mode processes have quit.
init check memory pass.
kernel panic at kern/process/proc.c:464:
    initproc exit.

os12011702@vmos-tony:~/oslab/lab12/Lab12$
```