

OS HW2 Report 0440062 黃濤

1. 四種方法執行時間結果如下（2，4，8，16 threads 的順序）：

mutex:

```
tobb@tobb-VirtualBox:~/桌面/mutex/mutex$ ./main 2
200000000
time: 4.995986s
```

```
tobb@tobb-VirtualBox:~/桌面/mutex/mutex$ ./main 4
400000000
time: 9.819202s
```

```
tobb@tobb-VirtualBox:~/桌面/mutex/mutex$ ./main 8
800000000
time: 19.844807s
```

```
tobb@tobb-VirtualBox:~/桌面/mutex/mutex$ ./main 16
1600000000
time: 39.090485s
```

sem:

```
tobb@tobb-VirtualBox:~/桌面/sem/sem$ ./main 2
200000000
time: 0.501313s
```

```
tobb@tobb-VirtualBox:~/桌面/sem/sem$ ./main 4
400000000
time: 1.748741s
```

```
tobb@tobb-VirtualBox:~/桌面/sem/sem$ ./main 8
800000000
time: 6.308232s
```

```
tobb@tobb-VirtualBox:~/桌面/sem/sem$ ./main 16
1600000000
time: 22.961087s
```

spinlock:

```
tobb@tobb-VirtualBox:~/桌面/spinlock/spinlock$ ./main 2
20000000
time: 0.344478s

tobb@tobb-VirtualBox:~/桌面/spinlock/spinlock$ ./main 4
40000000
time: 1.323311s

tobb@tobb-VirtualBox:~/桌面/spinlock/spinlock$ ./main 8
80000000
time: 5.108070s

tobb@tobb-VirtualBox:~/桌面/spinlock/spinlock$ ./main 16
160000000
time: 17.464079s
```

homemade_spinlock:

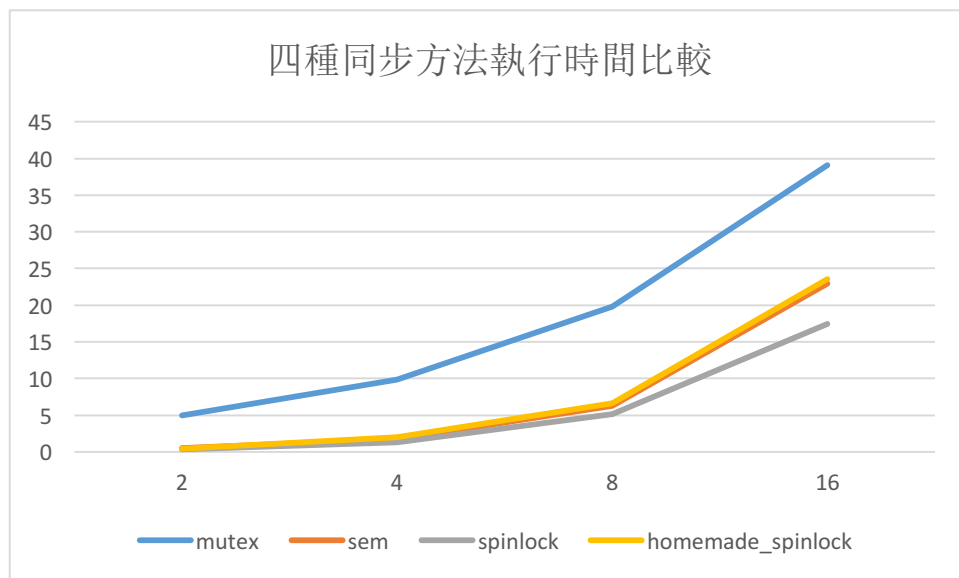
```
tobb@tobb-VirtualBox:~/桌面/homemade_spinlock/homemade_spinlock$ ./main 2
20000000
time: 0.456203s

tobb@tobb-VirtualBox:~/桌面/homemade_spinlock/homemade_spinlock$ ./main 4
40000000
time: 2.012579s

tobb@tobb-VirtualBox:~/桌面/homemade_spinlock/homemade_spinlock$ ./main 8
80000000
time: 6.626944s

tobb@tobb-VirtualBox:~/桌面/homemade_spinlock/homemade_spinlock$ ./main 16
160000000
time: 23.534574s
```

綜上，得到如下表



得到結論：

Pthread_mutex 執行時間最長，Semaphore 和 Homemade_spinlock 執行時間基本一致，Pthread_spinlock 執行時間最短。

2. 兩種方法執行時間如下：

(lock free)

```
tobb@tobb-VirtualBox:~/桌面/MonteCarlo1/MonteCarlo1$ ./main 100000000  
3.142073  
time: 2.779594s
```

(use lock)

```
tobb@tobb-VirtualBox:~/桌面/MonteCarlo2/MonteCarlo2$ ./main 100000000  
3.141797  
time: 5.204577s
```

由此可見，使用 lock 之後的執行時間更長，大概是 lock free 所需時間的兩倍。

3. 由題意分析可知，產生死鎖的唯一情況是四個線程相繼申請第二把鎖失敗（不必在同一個 time slot），因此我在 code 中開了一個長度為 4 的正列用於記錄 4 個 thread 最近一次申請第二把鎖是否失敗。當某一線程檢測到 4 個線程最近一次都申請失敗（即正列的四個 value 都為 1），則死鎖已發生。

死鎖發生后，立刻釋放檢測到死鎖的線程所持有的第一把鎖，并 rollback 到申請第一把鎖的 step。死鎖解除。此時有兩種可能情況：

1. 某一個線程在同一 time slot 內拿到剛剛釋放的鎖； 2. 該 time slot 無線程拿到剛剛釋放的鎖，則有可能再次發生 deadlock，但是再次發生的機率很低。