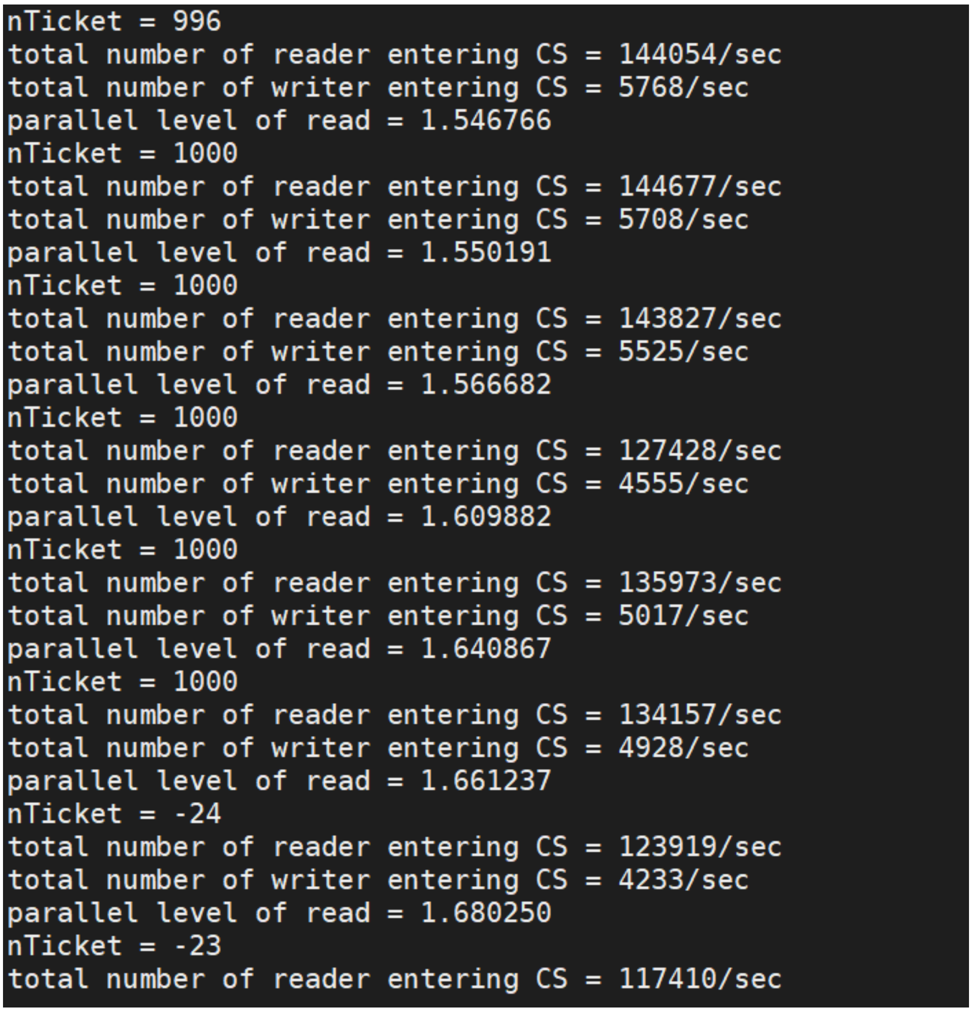
作業系統作業 8

經濟三 407510046 黃郁雯

1.



```
nTicket = 996
total number of reader entering CS = 144054/sec
total number of writer entering CS = 5768/sec
parallel level of read = 1.546766
nTicket = 1000
total number of reader entering CS = 144677/sec
total number of writer entering CS = 5708/sec
parallel level of read = 1.550191
nTicket = 1000
total number of reader entering CS = 143827/sec
total number of writer entering CS = 5525/sec
parallel level of read = 1.566682
nTicket = 1000
total number of reader entering CS = 127428/sec
total number of writer entering CS = 4555/sec
parallel level of read = 1.609882
nTicket = 1000
total number of reader entering CS = 135973/sec
total number of writer entering CS = 5017/sec
parallel level of read = 1.640867
nTicket = 1000
total number of reader entering CS = 134157/sec
total number of writer entering CS = 4928/sec
parallel level of read = 1.661237
nTicket = -24
total number of reader entering CS = 123919/sec
total number of writer entering CS = 4233/sec
parallel level of read = 1.680250
nTicket = -23
total number of reader entering CS = 117410/sec
```

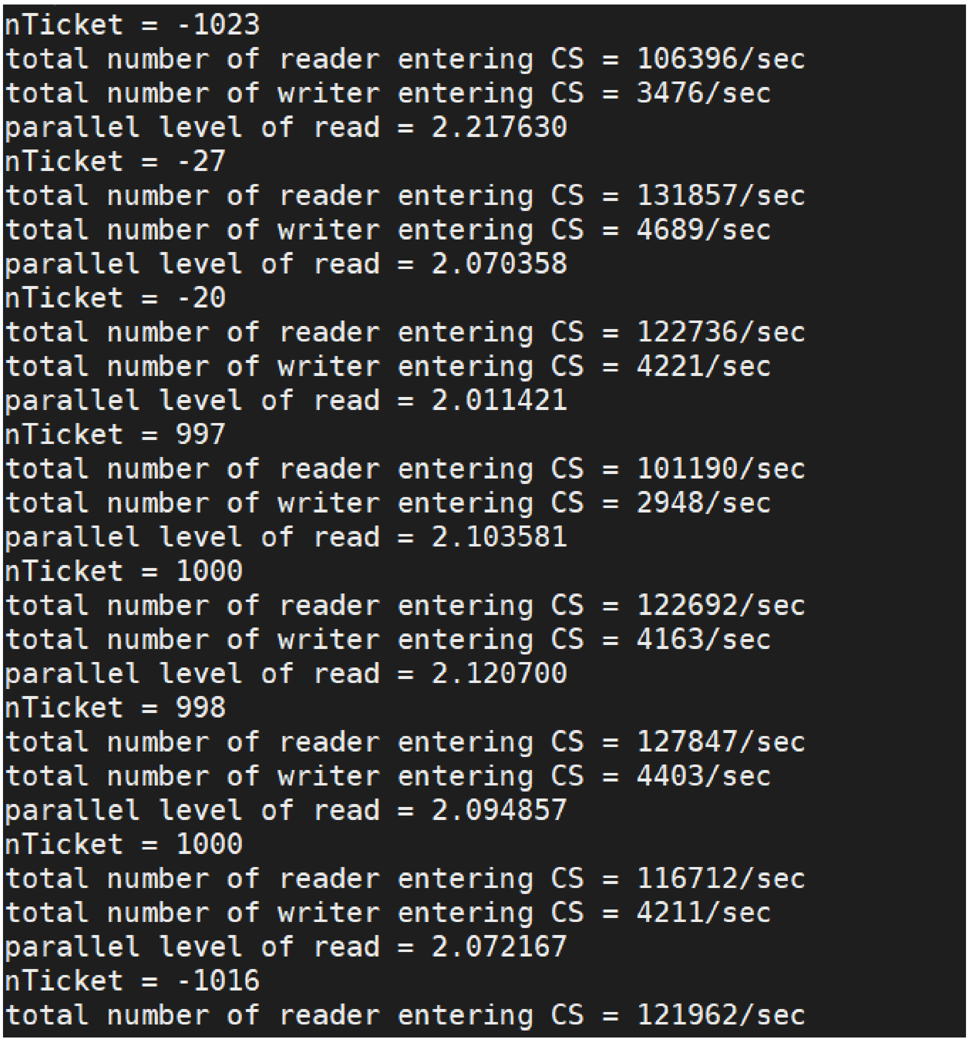
圖片為執行(./rwlock_w_order 500 550)結果。

做法為當 reader 進入 critical section 就減少一張門票，writer 則會減少 1000 張門票，因此，以上面圖片顯示出：假設 nTicket 為 996，則代表有 9 個 reader 在 critical section.

因為一共只有 1000 張門票，因此，reader 比 writer 有比較高的優先權。

而關於平行度，以最後一個 nTicket = 996 來說，parallel level of read =1.546766，代表只要有 reader 在 critical section 中，通常是 1.546766 個 reader 在 critical section 中。

2.



```
nTicket = -1023
total number of reader entering CS = 106396/sec
total number of writer entering CS = 3476/sec
parallel level of read = 2.217630
nTicket = -27
total number of reader entering CS = 131857/sec
total number of writer entering CS = 4689/sec
parallel level of read = 2.070358
nTicket = -20
total number of reader entering CS = 122736/sec
total number of writer entering CS = 4221/sec
parallel level of read = 2.011421
nTicket = 997
total number of reader entering CS = 101190/sec
total number of writer entering CS = 2948/sec
parallel level of read = 2.103581
nTicket = 1000
total number of reader entering CS = 122692/sec
total number of writer entering CS = 4163/sec
parallel level of read = 2.120700
nTicket = 998
total number of reader entering CS = 127847/sec
total number of writer entering CS = 4403/sec
parallel level of read = 2.094857
nTicket = 1000
total number of reader entering CS = 116712/sec
total number of writer entering CS = 4211/sec
parallel level of read = 2.072167
nTicket = -1016
total number of reader entering CS = 121962/sec
```

加上memory order 之後，效率提高。平行度相較於沒有memory order增加。

因為為了提高效能，編譯器會對程式碼進行重新排序，而這樣在單線程下，通常不會有問題，但是在多線程中可能就會有問題。

memory_order_relaxed是保證當前操作的原子性，不考慮線程的同步。