a12results.txt

```
##############################################

-------------------
Timed Test (1 thread)
------------------------------------------------------------
[1m[32mCS 218 - Assignment #12[0m

[1mDuck Numbers Counting Program[0m

Thread Count: 1
Numbers Limit: 400000000

[1m   Start Counting...[0m
 ...Thread starting...

[1mResults:[0m
--------
Duck Number Count: 151784532

-------------------
Timed Test (2 thread)
------------------------------------------------------------
[1m[32mCS 218 - Assignment #12[0m

[1mDuck Numbers Counting Program[0m

Thread Count: 2
Numbers Limit: 400000000

[1m   Start Counting...[0m
 ...Thread starting...
 ...Thread starting...

[1mResults:[0m
--------
Duck Number Count: 151784532

-------------------
Timed Test (3 thread)
------------------------------------------------------------
[1m[32mCS 218 - Assignment #12[0m

[1mDuck Numbers Counting Program[0m
```

```
Thread Count: 3
Numbers Limit: 400000000

[1m   Start Counting...[0m
 ...Thread starting...
 ...Thread starting...
 ...Thread starting...

[1mResults:[0m
--------
Duck Number Count: 151784532

-------------------
Timed Test (4 thread)
--------------------------------------------------------
[1m[32mCS 218 - Assignment #12[0m

[1mDuck Numbers Counting Program[0m

Thread Count: 4
Numbers Limit: 400000000

[1m   Start Counting...[0m
 ...Thread starting...
 ...Thread starting...
 ...Thread starting...
 ...Thread starting...

[1mResults:[0m
--------
Duck Number Count: 151784532
```

---

a12times.txt

```
##################################################

-------------------
Timed Test (1 thread)

real    0m7.440s
user    0m6.870s
sys     0m0.531s
```

Timed Test (2 thread)

```
real    0m4.511s
user    0m7.449s
sys     0m0.188s
```

Timed Test (3 thread)

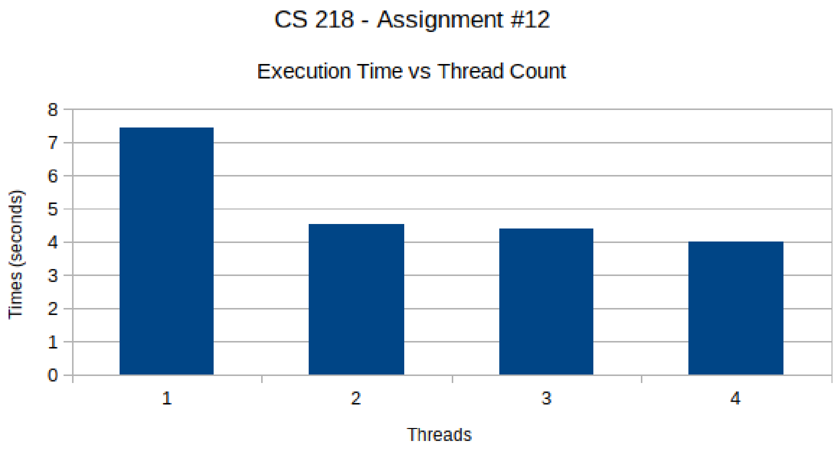
```
real    0m4.407s
user    0m7.524s
sys     0m0.158s
```

Timed Test (4 thread)

```
real    0m4.019s
user    0m7.517s
sys     0m0.194s
```

Speed-up Factors:
The speed up factors for mine are shown below using the base sequential execution and the parallel execution times formula.

Thread 2: 1.64930170693859
Thread 3: 1.68822328114364
Thread 4: 1.8512067678527



Duck Number Count (1 Thread and 4 Threads Executions):

1 Thread Execution:
First Run: Duck Number Count: 151784532

4 Thread Execution (multiple runs):
First Run: Duck Number Count: 151780401
Second Run: Duck Number Count: 151790607
Third Run: Duck Number Count: 151787448
Fourth Run: Duck Number Count: 151781373

Explanation:

The changes from the first thread execution to the four thread executions demonstrate that race conditions lead to inconsistent results. When the spinLock(), spinUnlock(), and the lock prefix were removed, these threads accessing the duckNumberCount variable led to these changes of the duck number count. As shown above, the first run to the fourth run were consistently changing with the limit of 1025602754. This highlights why spinLock() and spinUnlock() are essential. They help the program manage shared variables like duckNumberCount properly, stopping race conditions from occurring and false results. In terms of my chart, my first thread was the slowest time span where the occurrence of threads led to a gradual faster time.