Siddhartha (001084614) | Program Structures & Algorithms
Spring 2021 Assignment No. 2

TASK: To time insertion sort in 4 scenarios (random array, sorted array, partially sorted array and reverse ordered array)

OUTPUT:

```
2021-02-02 21:22:58 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:22:58 INFO  Benchmark_Timer - Value of N:5 Time to sort random array :0.00314 ms
2021-02-02 21:22:58 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:22:58 INFO  Benchmark_Timer - Value of N:25 Time to sort random array :0.03121 ms
2021-02-02 21:22:58 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:22:58 INFO  Benchmark_Timer - Value of N:125 Time to sort random array :0.26957 ms
2021-02-02 21:22:58 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:22:58 INFO  Benchmark_Timer - Value of N:625 Time to sort random array :0.93941 ms
2021-02-02 21:22:58 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:22:59 INFO  Benchmark_Timer - Value of N:3125 Time to sort random array :7.86517 ms
2021-02-02 21:22:59 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:23:01 INFO  Benchmark_Timer - Value of N:15625 Time to sort random array :181.73396 ms
2021-02-02 21:23:01 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:23:01 INFO  Benchmark_Timer - Value of N:5 Time to sort Partially Sorted Array:0.00079 ms
2021-02-02 21:23:01 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:23:01 INFO  Benchmark_Timer - Value of N:25 Time to sort Partially Sorted Array:0.00122 ms
2021-02-02 21:23:01 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:23:01 INFO  Benchmark_Timer - Value of N:125 Time to sort Partially Sorted Array:0.01651 ms
2021-02-02 21:23:01 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:23:01 INFO  Benchmark_Timer - Value of N:625 Time to sort Partially Sorted Array:0.23286 ms
2021-02-02 21:23:01 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:23:01 INFO  Benchmark_Timer - Value of N:3125 Time to sort Partially Sorted Array:5.67459 ms
2021-02-02 21:23:01 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:23:03 INFO  Benchmark_Timer - Value of N:15625 Time to sort Partially Sorted Array:142.58208 ms
2021-02-02 21:23:03 INFO  Benchmark_Timer - Begin run: Timing Insertion sort with 10 runs
2021-02-02 21:23:03 INFO  Benchmark_Timer - Value of N:5 Time to sort Ordered Array :0.00061 ms
```

Relationship Conclusion:

The order of growth of insertion sort is $O(N^2)$

Evidence to support the conclusion:

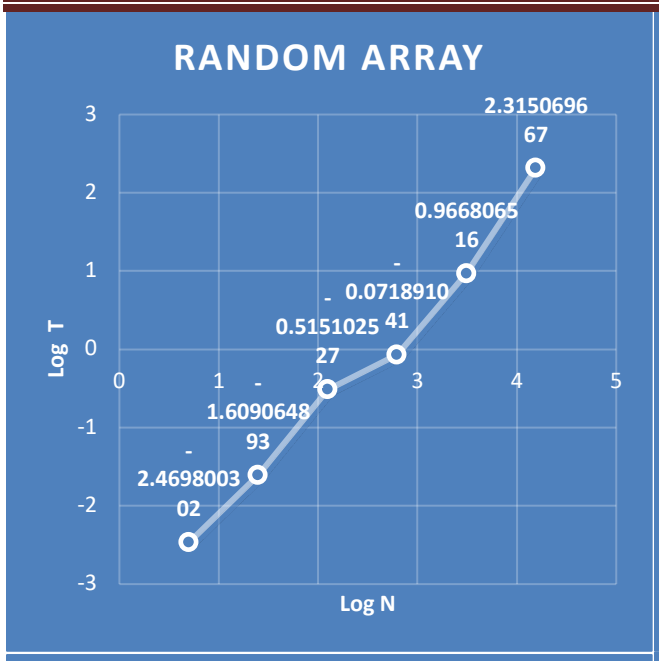
1) I ran insertion sort for 6 different array size of 5,25,125,625,3125,15625
2) I ran the insertion sort method 10 times with 2 warmups for each different size and obtained the following data

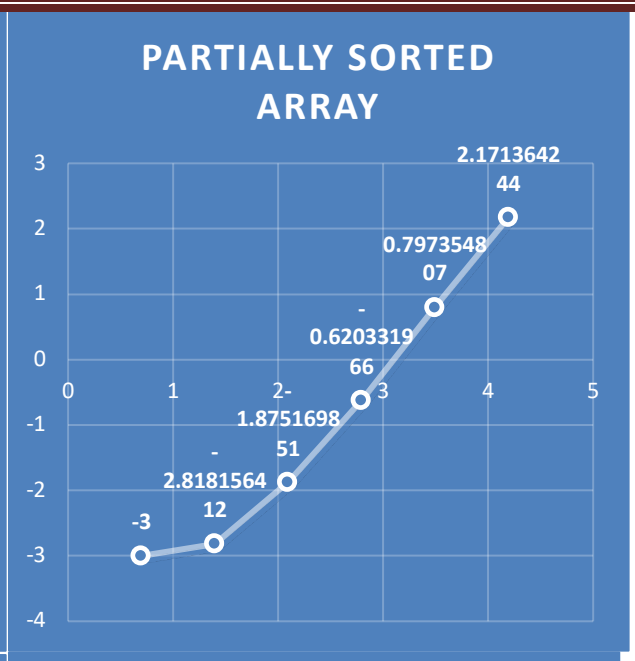
Type of Array	Value of N	Time(ms)	logn	logt
random array	5	0.00339	0.698970004	-2.4698003
random array	25	0.0246	1.397940009	-1.6090649
random array	125	0.30542	2.096910013	-0.5151025
random array	625	0.84744	2.795880017	-0.071891
random array	3125	9.26417	3.494850022	0.96680652
random array	15625	206.57115	4.193820026	2.31506967
Partially Sorted	5	0.001	0.698970004	-3
Partially Sorted	25	0.00152	1.397940009	-2.8181564
Partially Sorted	125	0.01333	2.096910013	-1.8751699

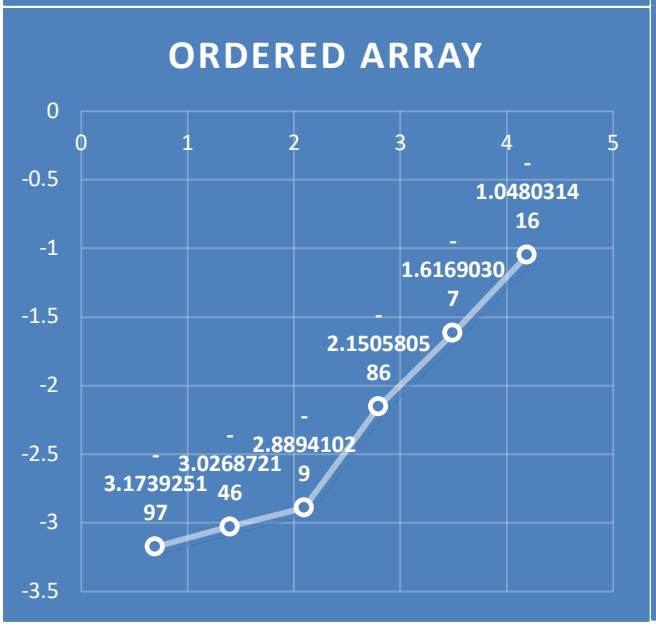
Partially Sorted	625	0.2397	2.795880017	-0.620332
Partially Sorted	3125	6.27126	3.494850022	0.79735481
Partially Sorted	15625	148.3762	4.193820026	2.17136424
Ordered Array	5	0.00067	0.698970004	-3.1739252
Ordered Array	25	0.00094	1.397940009	-3.0268721
Ordered Array	125	0.00129	2.096910013	-2.8894103
Ordered Array	625	0.00707	2.795880017	-2.1505806
Ordered Array	3125	0.02416	3.494850022	-1.6169031
Ordered Array	15625	0.08953	4.193820026	-1.0480314
Reverse Order	5	0.00222	0.698970004	-2.653647
Reverse Order	25	0.00206	1.397940009	-2.6861328
Reverse Order	125	0.03144	2.096910013	-1.5025175
Reverse Order	625	0.72158	2.795880017	-0.1417155
Reverse Order	3125	17.90089	3.494850022	1.25287462
Reverse Order	15625	406.63773	4.193820026	2.60920767

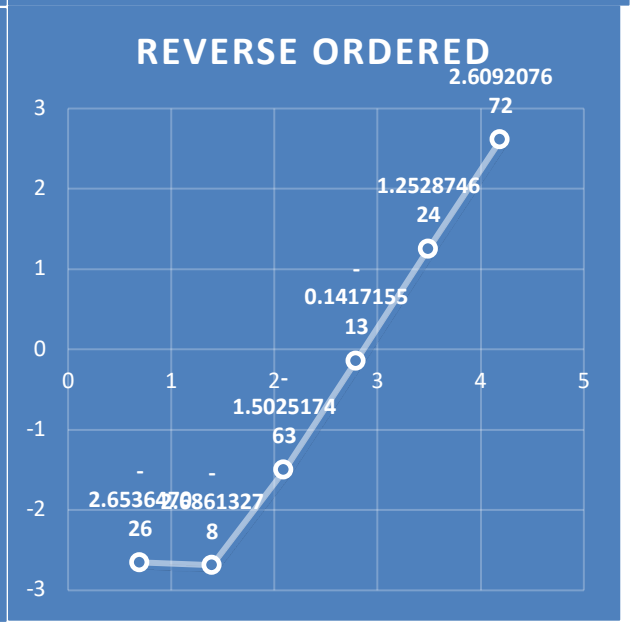
3) Based on the data obtained I plotted a graph between logN(X-axis) and LogT(y-axis)
4) Slope of the graph is approx 1.928 which is order of $O(N^2)$

Graphical representation:

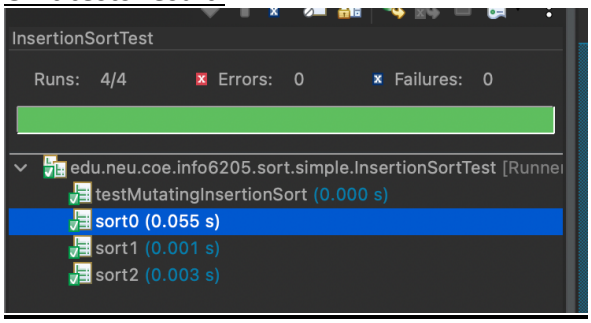


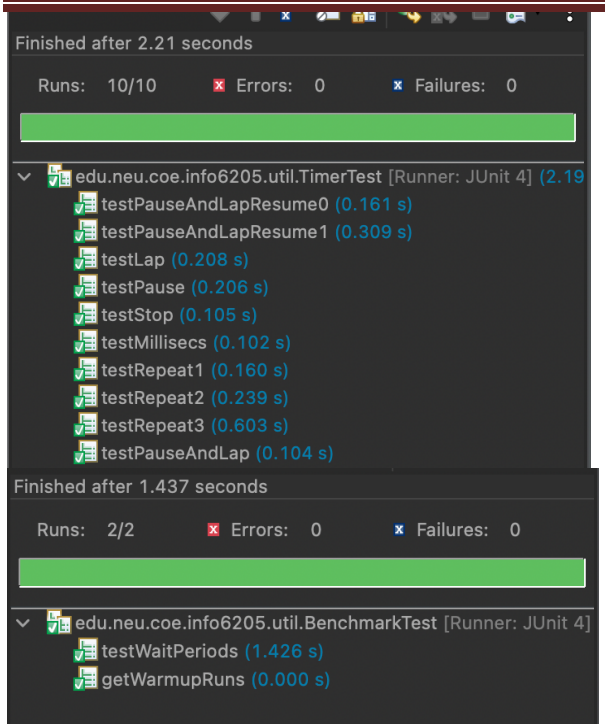






Unit tests result:




Finished after 2.21 seconds
Runs: 10/10
Errors: 0
Failures: 0
edu.neu.coe.info6205.util.TimerTest [Runner: JUnit 4] (2.19
testPauseAndLapResume0 (0.161 s)
testPauseAndLapResume1 (0.309 s)
testLap (0.208 s)
testPause (0.206 s)
testStop (0.105 s)
testMillisecs (0.102 s)
testRepeat1 (0.160 s)
testRepeat2 (0.239 s)
testRepeat3 (0.603 s)
testPauseAndLap (0.104 s)
Finished after 1.437 seconds
Runs: 2/2
Errors: 0
Failures: 0
edu.neu.coe.info6205.util.BenchmarkTest [Runner: JUnit 4]
testWaitPeriods (1.426 s)
getWarmupRuns (0.000 s)