Gan Li (001063585)

Program Structures & Algorithms

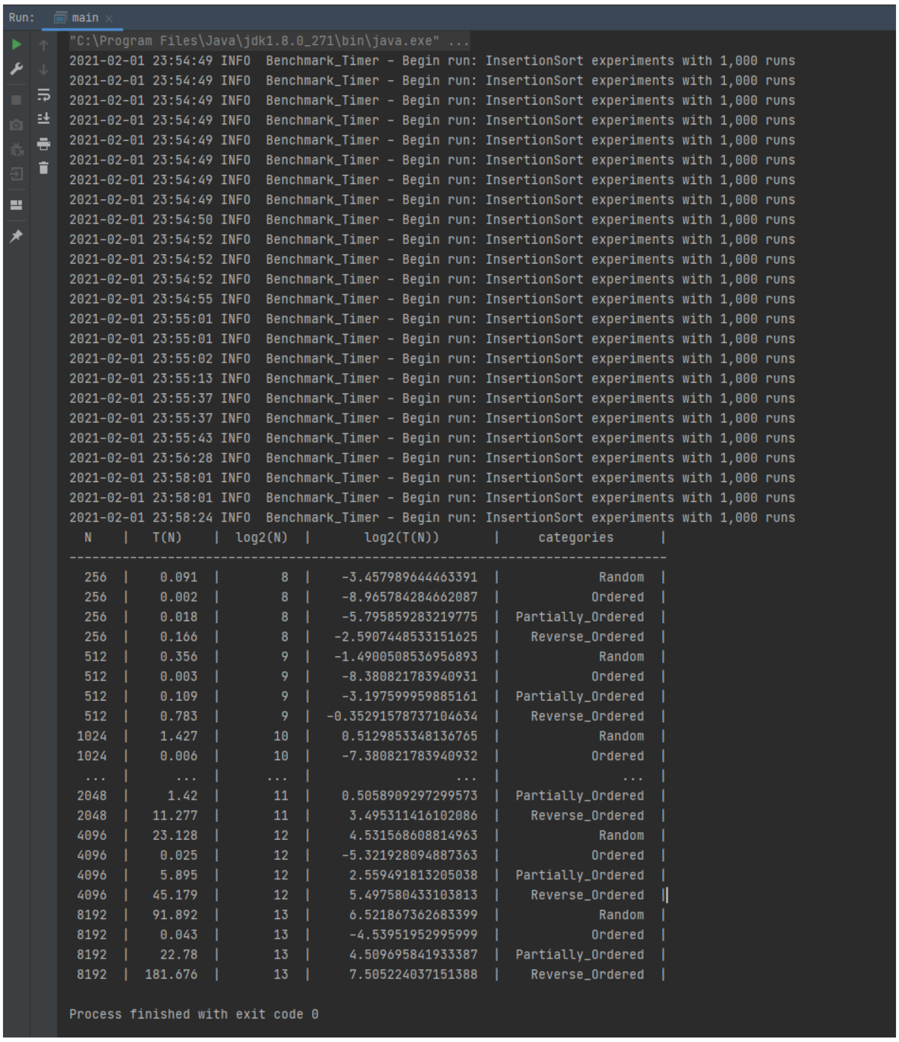
Assignment NO.2

1. Task

- Implement three methods of a class called Timer.
- Implement **sort** method in class InsertionSort.
- Measure the running times of insertion sort by different initial array ordering situations and values of n (number of elements in array).
- Draw conclusions about the order of growth from those experiments.

2. Output

6 values (from 256 to 8192) of n and 1000-times running of each n.



```
"C:\Program Files\Java\jdk1.8.0_271\bin\java.exe" ...
2021-02-01 23:54:49 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:49 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:49 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:49 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:49 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:49 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:49 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:49 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:50 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:52 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:52 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:52 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:54:55 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:55:01 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:55:01 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:55:02 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:55:13 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:55:37 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:55:37 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:55:43 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:56:28 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:58:01 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:58:01 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
2021-02-01 23:58:24 INFO  Benchmark_Timer - Begin run: InsertionSort experiments with 1,000 runs
  N    |   T(N)    |  log2(N)  |        log2(T(N))       |     categories      |
-------------------------------------------------------------------------------
  256  |    0.091  |        8  |    -3.457989644463391  |            Random  |
  256  |    0.002  |        8  |    -8.965784284662087  |           Ordered  |
  256  |    0.018  |        8  |    -5.795859283219775  |  Partially_Ordered  |
  256  |    0.166  |        8  |   -2.5907448533151625  |    Reverse_Ordered  |
  512  |    0.356  |        9  |   -1.4900508536956893  |            Random  |
  512  |    0.003  |        9  |    -8.380821783940931  |           Ordered  |
  512  |    0.109  |        9  |    -3.197599959885161  |  Partially_Ordered  |
  512  |    0.783  |        9  |  -0.35291578737104634  |    Reverse_Ordered  |
 1024  |    1.427  |       10  |    0.5129853348136765  |            Random  |
 1024  |    0.006  |       10  |    -7.380821783940932  |           Ordered  |
  ...  |      ...  |      ...  |                   ...  |               ...  |
 2048  |     1.42  |       11  |    0.505890929729957З  |  Partially_Ordered  |
 2048  |   11.277  |       11  |     3.495311416102086  |    Reverse_Ordered  |
 4096  |   23.128  |       12  |     4.531568608814963  |            Random  |
 4096  |    0.025  |       12  |    -5.321928094887363  |           Ordered  |
 4096  |    5.895  |       12  |     2.559491813205038  |  Partially_Ordered  |
 4096  |   45.179  |       12  |     5.497580433103813  |    Reverse_Ordered  ||
 8192  |   91.892  |       13  |     6.521867362683399  |            Random  |
 8192  |    0.043  |       13  |    -4.53951952995999  |           Ordered  |
 8192  |    22.78  |       13  |     4.509695841933387  |  Partially_Ordered  |
 8192  |  181.676  |       13  |     7.505224037151388  |    Reverse_Ordered  |

Process finished with exit code 0
```

3. Relationship Conclusion

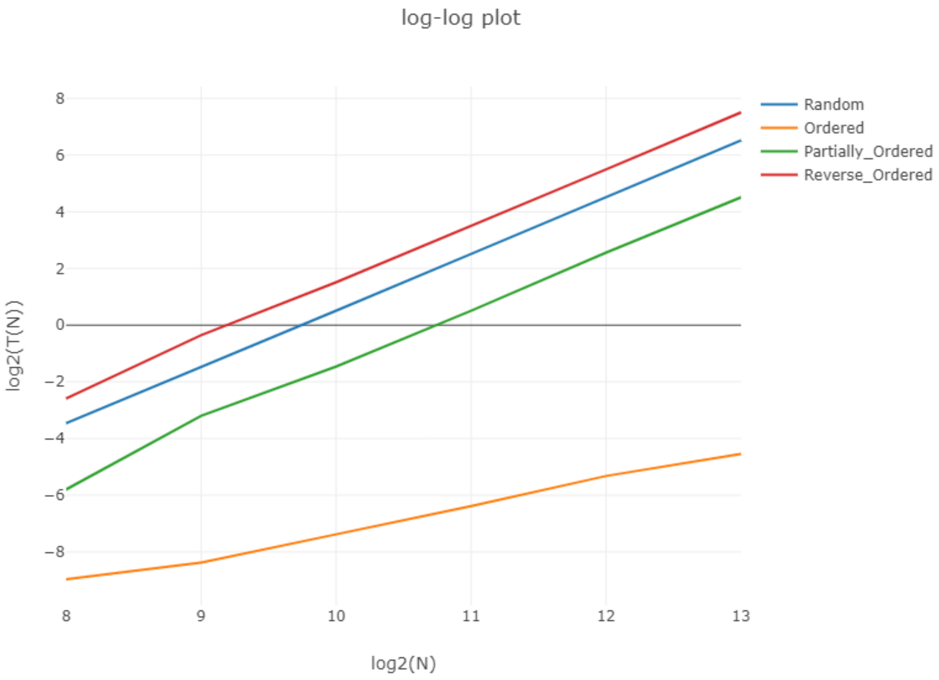
The initial array ordering situation affects the running time of insertion sort. When the initial array is reverse ordered, it increases sorting time most, and then is random, partially ordered and ordered array.

The time complexity of insertion sort is $O(N^2)$ when the initial array is random, partially ordered, reverse ordered. It becomes $O(N)$ when the initial array is ordered.

4. Evidence to support the conclusion:

- Log-log plot

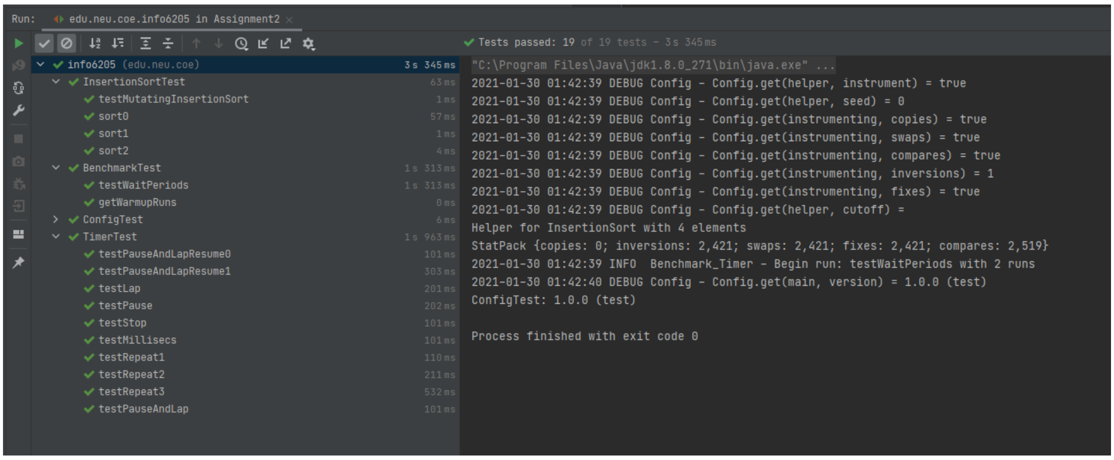
As shown below, whether the initial array is random, partially ordered, or reverse ordered①, their results are all a straight line with similar slope. And when the initial array is ordered②, its line becomes flatter. In a log-log plot, the slope represents the power-law relationship between x and y. Moreover, as data shown in table below, in situation 1 running time becomes 4 times when n doubled, 2 times in situation 2. Therefore, $\mathrm{T(N)}{\sim}\mathrm{N}^2$ when in situation 1, $\mathrm{T(N)}{\sim}N$ in situation 2.



- Table (full version is under zip file named **Results**)

T(N)/s N	Random	Ordered	Part-Ordered	Reverse-Ordered
256	0.091	0.002	0.018	0.166
512	0.356	0.003	0.109	0.783
1024	1.427	0.006	0.363	2.859
2048	5.712	0.012	1.42	11.277
4096	23.128	0.025	5.895	45.179
8192	91.892	0.043	22.78	181.676

5. Unit test results



```
✓ Tests passed: 19 of 19 tests – 3 s 345 ms
```

```
✓ info6205 (edu.neu.coe)                3 s 345 ms
  ✓ InsertionSortTest                        63 ms
    ✓ testMutatingInsertionSort               1 ms
    ✓ sort0                                  57 ms
    ✓ sort1                                   1 ms
    ✓ sort2                                   4 ms
  ✓ BenchmarkTest                       1 s 313 ms
    ✓ testWaitPeriods                   1 s 313 ms
    ✓ getWarmupRuns                           0 ms
  ✓ ConfigTest                                6 ms
  ✓ TimerTest                           1 s 963 ms
    ✓ testPauseAndLapResume0                101 ms
    ✓ testPauseAndLapResume1                303 ms
    ✓ testLap                               201 ms
    ✓ testPause                             202 ms
    ✓ testStop                              101 ms
    ✓ testMillisecs                         101 ms
    ✓ testRepeat1                           110 ms
    ✓ testRepeat2                           211 ms
    ✓ testRepeat3                           532 ms
    ✓ testPauseAndLap                       101 ms
```

```
"C:\Program Files\Java\jdk1.8.0_271\bin\java.exe" ...
2021-01-30 01:42:39 DEBUG Config - Config.get(helper, instrument) = true
2021-01-30 01:42:39 DEBUG Config - Config.get(helper, seed) = 0
2021-01-30 01:42:39 DEBUG Config - Config.get(instrumenting, copies) = true
2021-01-30 01:42:39 DEBUG Config - Config.get(instrumenting, swaps) = true
2021-01-30 01:42:39 DEBUG Config - Config.get(instrumenting, compares) = true
2021-01-30 01:42:39 DEBUG Config - Config.get(instrumenting, inversions) = 1
2021-01-30 01:42:39 DEBUG Config - Config.get(instrumenting, fixes) = true
2021-01-30 01:42:39 DEBUG Config - Config.get(helper, cutoff) = 
Helper for InsertionSort with 4 elements
StatPack {copies: 0; inversions: 2,421; swaps: 2,421; fixes: 2,421; compares: 2,519}
2021-01-30 01:42:39 INFO  Benchmark_Timer - Begin run: testWaitPeriods with 2 runs
2021-01-30 01:42:40 DEBUG Config - Config.get(main, version) = 1.0.0 (test)
ConfigTest: 1.0.0 (test)

Process finished with exit code 0
```

6. Code

The project of this assignment is called **Assignment2** in the zip file.