

Yuhan Yang (001094267)

Program Structures & Algorithms

Fall 2021

Assignment No. 2

◉ **Task (List down the tasks performed in the Assignment)**

1. You are to implement three methods of a class called Timer. Please see the skeleton class that I created in the repository.
2. Implement InsertionSort (in the InsertionSort class) by simply looking up the insertion code used by Arrays.sort.
3. Implement a main program (or you could do it via your own unit tests) to actually run the following benchmarks: measure the running times of this sort, using four different initial array ordering situations: random, ordered, partially-ordered and reverse-ordered.

◉ **Conclusion:**

The ordered array takes the least time which is nearly 0.

The reversed array takes the longest time to be sorted.

The time of random array and partially-ordered array take is in the middle. The mean time of this two, depending on how much elements are in ordered, is close.

◉ **Evidence to support the conclusion:**

Output (Snapshot of Code output in the terminal)

```
Run with 'Windows Async Profiler': InsertionSortMain x
Mean Time for Reversed Array is: 0.37

Array Size: 400
Random array
2021-09-24 18:27:07 INFO Benchmark_Timer - Begin run: InsertionSort with 100 runs
Mean Time for Random Array is: 0.75
Ordered array
2021-09-24 18:27:07 INFO Benchmark_Timer - Begin run: InsertionSort with 100 runs
Mean Time for Ordered Array is: 0.0
Partially Ordered array
2021-09-24 18:27:07 INFO Benchmark_Timer - Begin run: InsertionSort with 100 runs
Mean Time for Partially Ordered Array is: 0.58
Reversed array
2021-09-24 18:27:07 INFO Benchmark_Timer - Begin run: InsertionSort with 100 runs
Mean Time for Reversed Array is: 1.61

Array Size: 800
Random array
2021-09-24 18:27:07 INFO Benchmark_Timer - Begin run: InsertionSort with 100 runs
Mean Time for Random Array is: 2.7
Ordered array
2021-09-24 18:27:07 INFO Benchmark_Timer - Begin run: InsertionSort with 100 runs
Mean Time for Ordered Array is: 0.01
Partially Ordered array
2021-09-24 18:27:07 INFO Benchmark_Timer - Begin run: InsertionSort with 100 runs
Mean Time for Partially Ordered Array is: 2.65
Reversed array
2021-09-24 18:27:08 INFO Benchmark_Timer - Begin run: InsertionSort with 100 runs
Mean Time for Reversed Array is: 6.93
```

◉ Unit tests result:(Snapshot of successful unit test run)

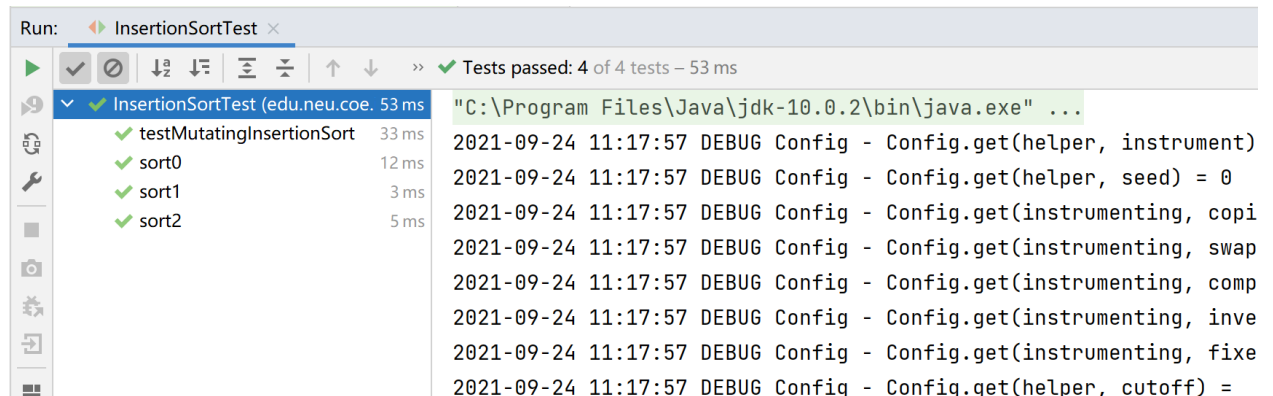
1. TimerTest

```
Run with 'Windows Async Profiler': TimerTest x
Tests passed: 10 of 10 tests - 2 sec 222 ms

TimerTest (edu.neu.coe.ir 2 sec 222 ms)
  ✓ testPauseAndLapResume0 372 ms
  ✓ testPauseAndLapResume1 302 ms
  ✓ testLap 202 ms
  ✓ testPause 203 ms
  ✓ testStop 100 ms
  ✓ testMillisecs 100 ms
  ✓ testRepeat1 108 ms
  ✓ testRepeat2 208 ms
  ✓ testRepeat3 525 ms
  ✓ testPauseAndLap 102 ms

"C:\Program Files\Java\jdk-10.0.2\bin\java.exe" ...
Started [cpu] profiling
Process finished with exit code 0
```

2. InsertionSortTest

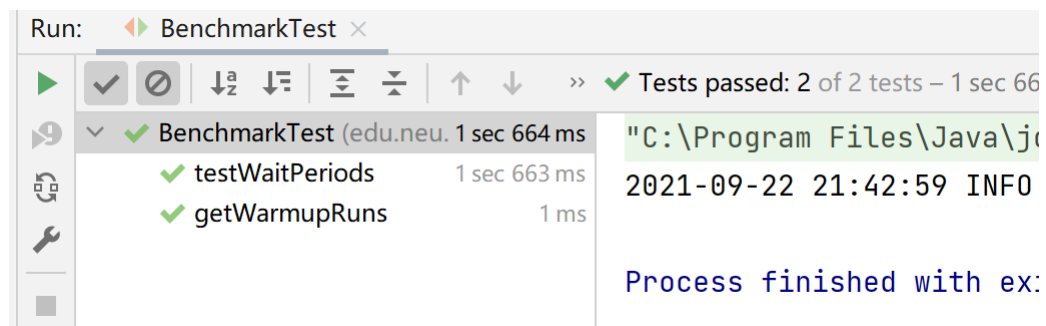


```
Run: InsertionSortTest x
Tests passed: 4 of 4 tests - 53 ms

InsertionSortTest (edu.neu.coe. 53 ms)
  testMutatingInsertionSort 33 ms
  sort0 12 ms
  sort1 3 ms
  sort2 5 ms

"C:\Program Files\Java\jdk-10.0.2\bin\java.exe" ...
2021-09-24 11:17:57 DEBUG Config - Config.get(helper, instrument)
2021-09-24 11:17:57 DEBUG Config - Config.get(helper, seed) = 0
2021-09-24 11:17:57 DEBUG Config - Config.get(instrumenting, copi
2021-09-24 11:17:57 DEBUG Config - Config.get(instrumenting, swap
2021-09-24 11:17:57 DEBUG Config - Config.get(instrumenting, comp
2021-09-24 11:17:57 DEBUG Config - Config.get(instrumenting, inve
2021-09-24 11:17:57 DEBUG Config - Config.get(instrumenting, fixe
2021-09-24 11:17:57 DEBUG Config - Config.get(helper, cutoff) =
```

3. BenchmarkTest



```
Run: BenchmarkTest x
Tests passed: 2 of 2 tests - 1 sec 664 ms

BenchmarkTest (edu.neu. 1 sec 664 ms)
  testWaitPeriods 1 sec 663 ms
  getWarmupRuns 1 ms

"C:\Program Files\Java\j
2021-09-22 21:42:59 INFO

Process finished with ex:
```