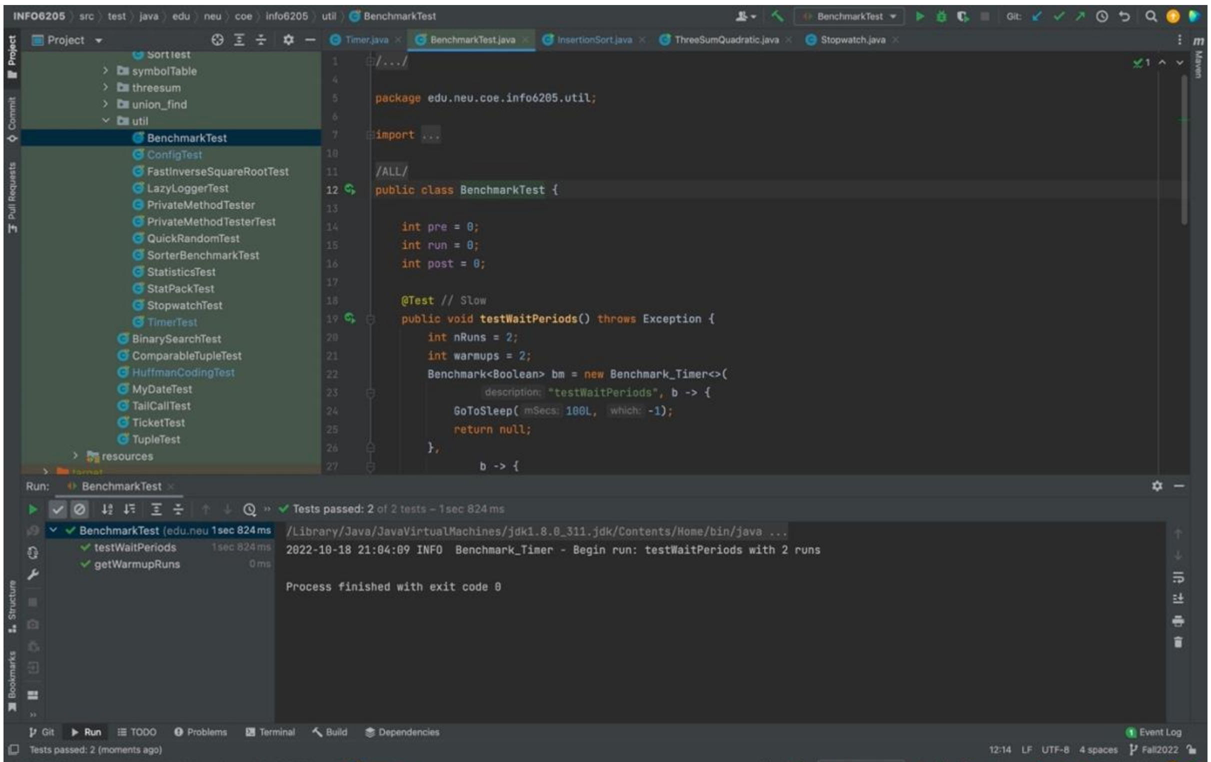
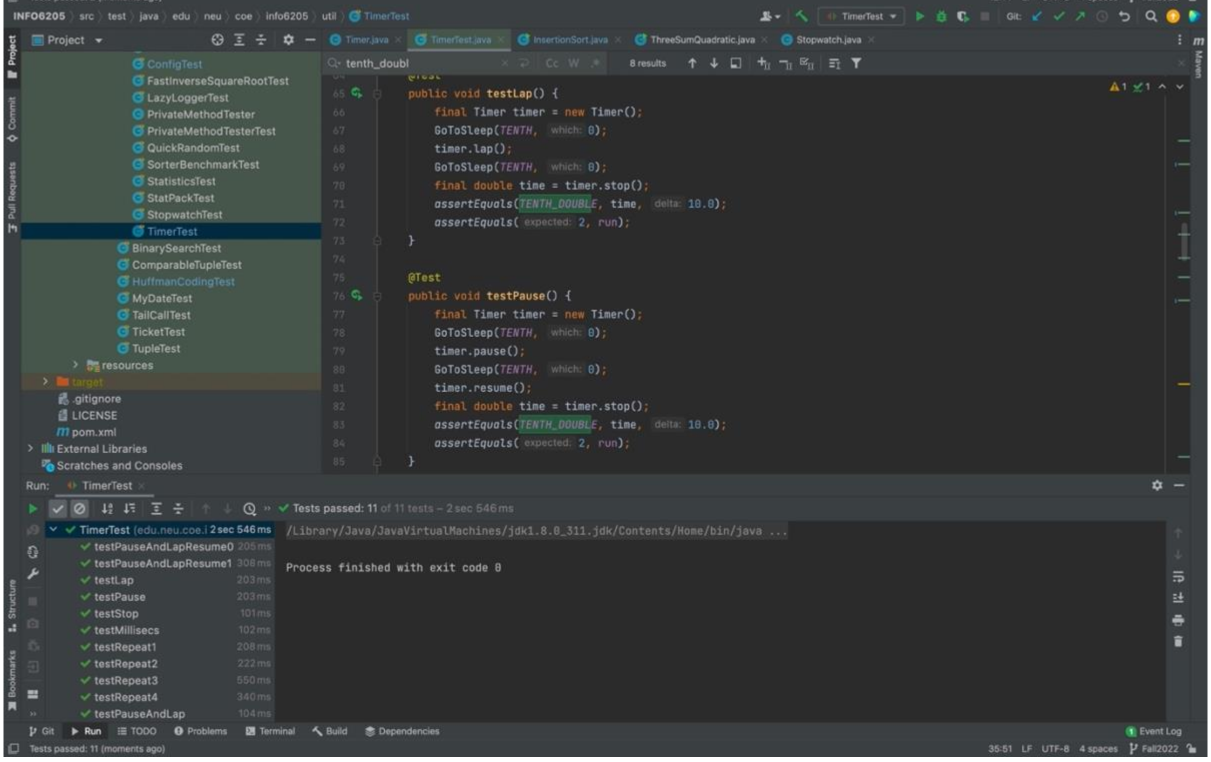
Assignment 3- Benchmarking

A Evidence



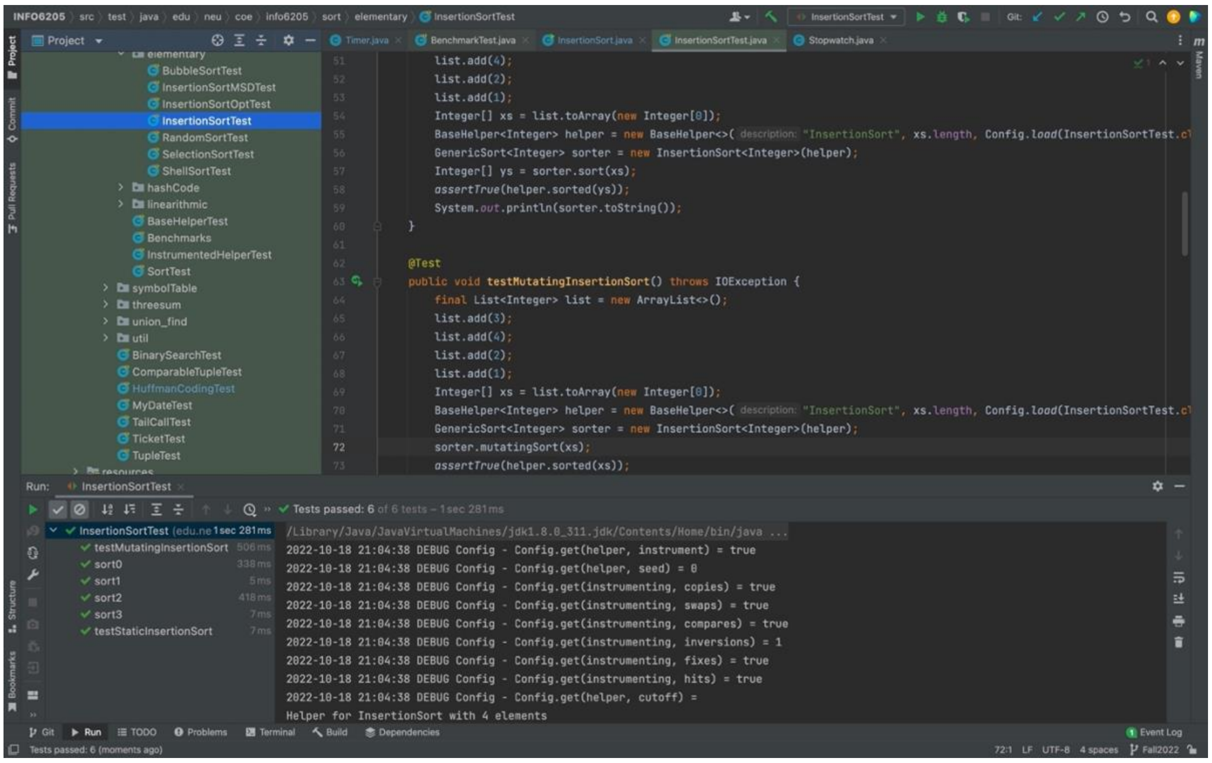
```java
/.../

package edu.neu.coe.info6205.util;

import ...

/ALL/
public class BenchmarkTest {

    int pre = 0;
    int run = 0;
    int post = 0;

    @Test // Slow
    public void testWaitPeriods() throws Exception {
        int nRuns = 2;
        int warmups = 2;
        Benchmark<Boolean> bm = new Benchmark_Timer<>(
                description: "testWaitPeriods", b -> {
            GoToSleep( mSecs: 100L, which: -1);
            return null;
        },
                b -> {
```

```
Tests passed: 2 of 2 tests – 1 sec 824 ms

BenchmarkTest (edu.neu 1 sec 824 ms
    testWaitPeriods    1 sec 824 ms
    getWarmupRuns      0 ms

/Library/Java/JavaVirtualMachines/jdk1.8.0_311.jdk/Contents/Home/bin/java ...
2022-10-18 21:04:09 INFO  Benchmark_Timer - Begin run: testWaitPeriods with 2 runs

Process finished with exit code 0
```



```java
    public void testLap() {
        final Timer timer = new Timer();
        GoToSleep(TENTH, which: 0);
        timer.lap();
        GoToSleep(TENTH, which: 0);
        final double time = timer.stop();
        assertEquals(TENTH_DOUBLE, time, delta: 10.0);
        assertEquals( expected: 2, run);
    }

    @Test
    public void testPause() {
        final Timer timer = new Timer();
        GoToSleep(TENTH, which: 0);
        timer.pause();
        GoToSleep(TENTH, which: 0);
        timer.resume();
        final double time = timer.stop();
        assertEquals(TENTH_DOUBLE, time, delta: 10.0);
        assertEquals( expected: 2, run);
    }
```

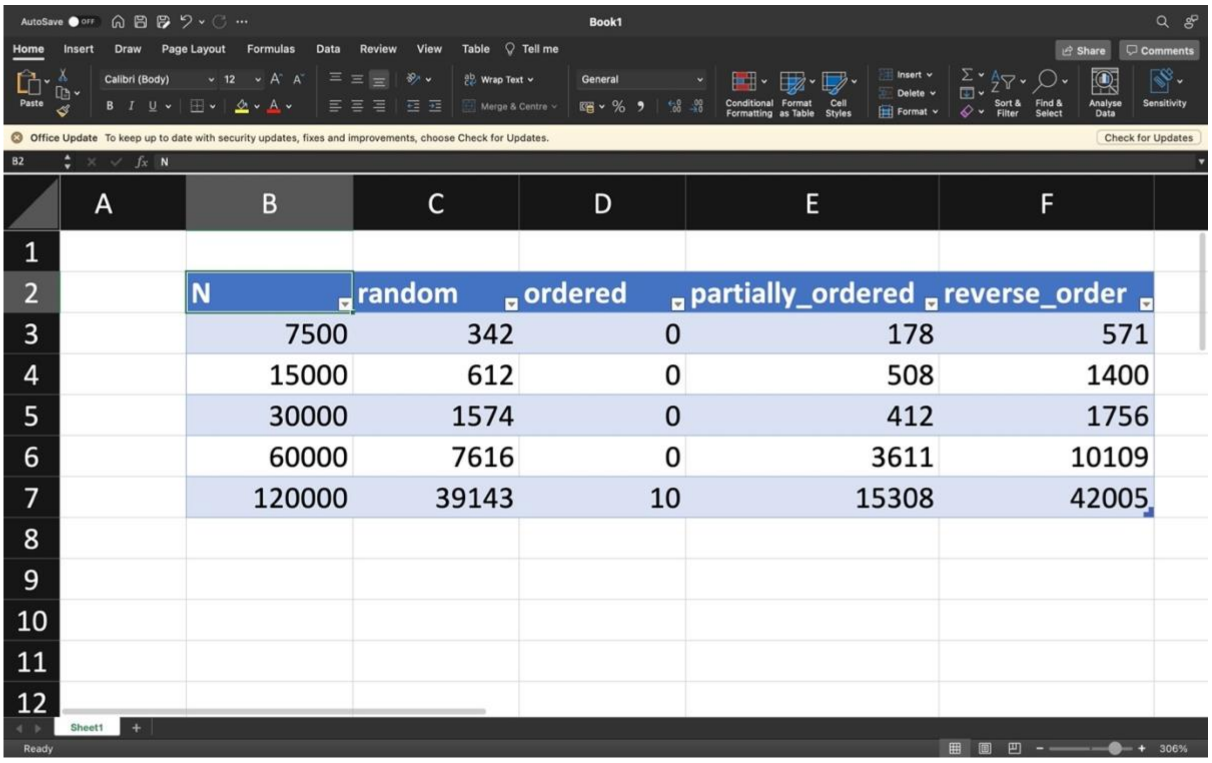
```
Tests passed: 11 of 11 tests – 2 sec 546 ms

TimerTest (edu.neu.coe.i 2 sec 546 ms
    testPauseAndLapResume0   205 ms
    testPauseAndLapResume1   308 ms
    testLap                  203 ms
    testPause                203 ms
    testStop                 101 ms
    testMillisecs            102 ms
    testRepeat1              208 ms
    testRepeat2              222 ms
    testRepeat3              550 ms
    testRepeat4              340 ms
    testPauseAndLap          104 ms

/Library/Java/JavaVirtualMachines/jdk1.8.0_311.jdk/Contents/Home/bin/java ...

Process finished with exit code 0
```



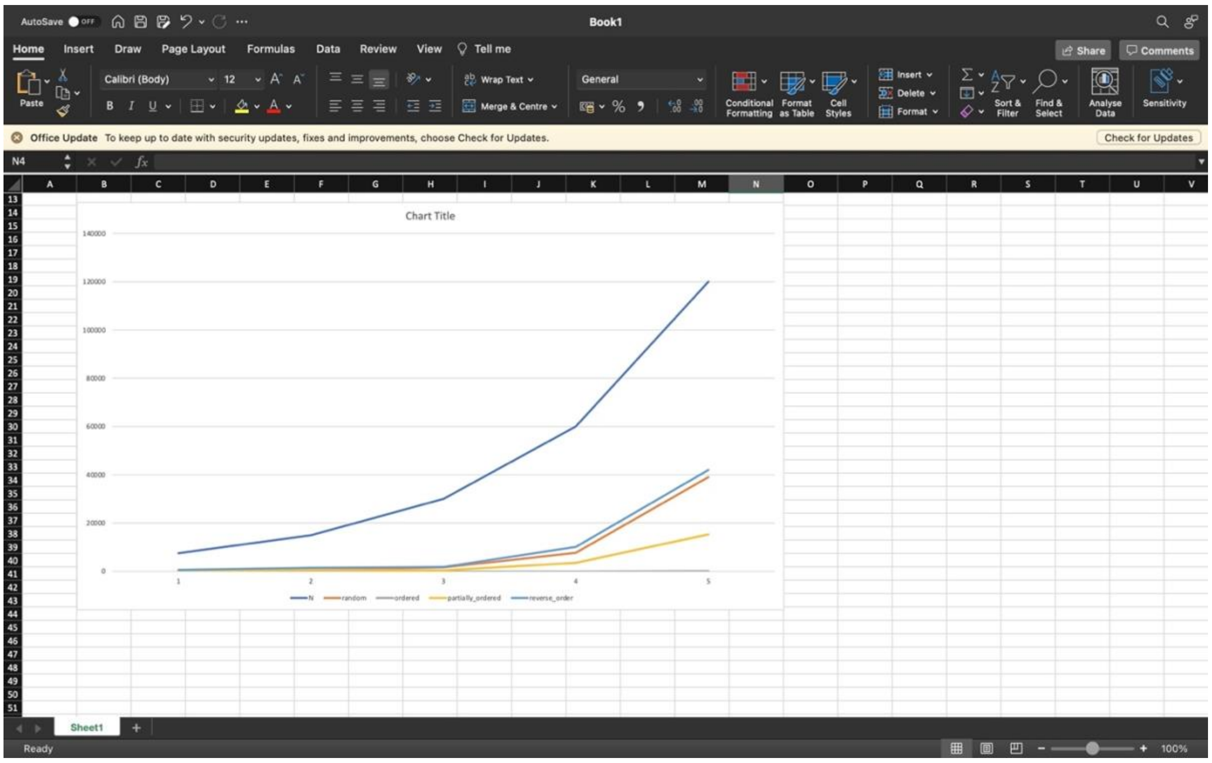
```java
        list.add(4);
        list.add(2);
        list.add(1);
        Integer[] xs = list.toArray(new Integer[0]);
        BaseHelper<Integer> helper = new BaseHelper<>( description: "InsertionSort", xs.length, Config.load(InsertionSortTest.cl
        GenericSort<Integer> sorter = new InsertionSort<Integer>(helper);
        Integer[] ys = sorter.sort(xs);
        assertTrue(helper.sorted(ys));
        System.out.println(sorter.toString());
    }

    @Test
    public void testMutatingInsertionSort() throws IOException {
        final List<Integer> list = new ArrayList<>();
        list.add(3);
        list.add(4);
        list.add(2);
        list.add(1);
        Integer[] xs = list.toArray(new Integer[0]);
        BaseHelper<Integer> helper = new BaseHelper<>( description: "InsertionSort", xs.length, Config.load(InsertionSortTest.cl
        GenericSort<Integer> sorter = new InsertionSort<Integer>(helper);
        sorter.mutatingSort(xs);
        assertTrue(helper.sorted(xs));
```

```
Tests passed: 6 of 6 tests – 1 sec 281 ms

InsertionSortTest (edu.ne 1 sec 281 ms
  testMutatingInsertionSort   506 ms
  sort0                       338 ms
  sort1                         5 ms
  sort2                       418 ms
  sort3                         7 ms
  testStaticInsertionSort       7 ms

/Library/Java/JavaVirtualMachines/jdk1.8.0_311.jdk/Contents/Home/bin/java ...
2022-10-18 21:04:38 DEBUG Config - Config.get(helper, instrument) = true
2022-10-18 21:04:38 DEBUG Config - Config.get(helper, seed) = 0
2022-10-18 21:04:38 DEBUG Config - Config.get(instrumenting, copies) = true
2022-10-18 21:04:38 DEBUG Config - Config.get(instrumenting, swaps) = true
2022-10-18 21:04:38 DEBUG Config - Config.get(instrumenting, compares) = true
2022-10-18 21:04:38 DEBUG Config - Config.get(instrumenting, inversions) = 1
2022-10-18 21:04:38 DEBUG Config - Config.get(instrumenting, fixes) = true
2022-10-18 21:04:38 DEBUG Config - Config.get(instrumenting, hits) = true
2022-10-18 21:04:38 DEBUG Config - Config.get(helper, cutoff) =
Helper for InsertionSort with 4 elements

Tests passed: 6 (moments ago)
```

B Observation



N	random	ordered	partially_ordered	reverse_order
7500	342	0	178	571
15000	612	0	508	1400
30000	1574	0	412	1756
60000	7616	0	3611	10109
120000	39143	10	15308	42005



C Relationship

Insertion sort compares adjacent elements in the array and swaps them if needed.

As a result, the ordered arrays take the least time because there are no inversions and minimal comparisons.

The reverse ordered arrays take the most time because the number of comparisons and the number of inversions are quadratic.

Partially ordered and random arrays are more dependent on the degree of sorting already available.