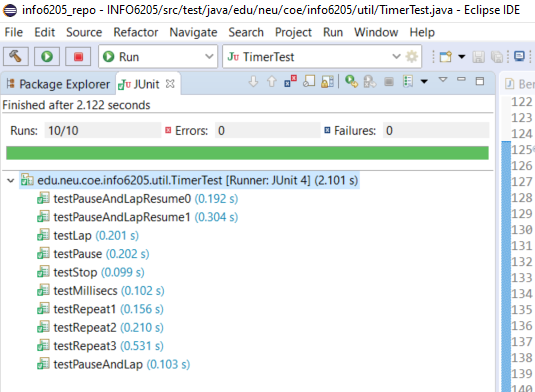
**INFO 6205**

**Program Structures & Algorithms**

**Fall 2020**

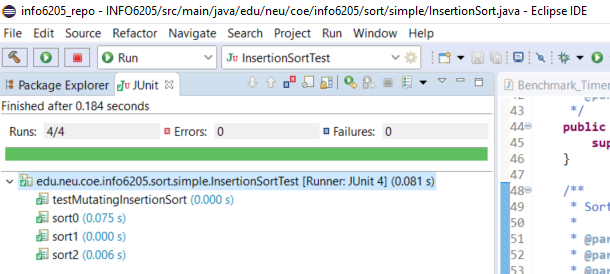
**Assignment 2**

* **Git repository:** <https://github.com/songyin79/INFO6205>
* **(Part 1) You are to implement four methods of a class called Timer.** 
  + [Timer.java](https://github.com/songyin79/INFO6205/blob/master/src/main/java/edu/neu/coe/info6205/util/Timer.java)
  + [Unit test](https://github.com/songyin79/INFO6205/blob/master/src/test/java/edu/neu/coe/info6205/util/TimerTest.java):



* **(Part 2) Implement InsertionSort (in the InsertionSort class) by simply looking up the insertion code used by Arrays.sort.**

* + [InsertionSort.java](https://github.com/songyin79/INFO6205/blob/master/src/main/java/edu/neu/coe/info6205/sort/simple/InsertionSort.java)
  + [Unit test](https://github.com/songyin79/INFO6205/blob/master/src/test/java/edu/neu/coe/info6205/sort/simple/InsertionSortTest.java):



* **(Part 3) Measure the running times of this sort, using four different initial array ordering situations: random, ordered, partially-ordered and reverse-ordered. I suggest that your arrays to be sorted are of type Integer. Use the doubling method for choosing n and test for at least five values of n. Draw any conclusions from your observations regarding the order of growth.**

According to the output.csv <https://github.com/songyin79/INFO6205/blob/master/output.csv>

the larger the size, the much time needed for sorting.

Reversed list takes most time than other type of lists.

Partially sorted list and sorted list take less time.

In a nutshell, the time needed for sorting is:

Reversed > random > partial >= sorted