**Program Structures & Algorithms**

**Spring 2022**

**Assignment No. N2**

Name: Shijie Zhang

(NUID): 001537250

* **Task**
* **Output screenshot**
* **Relationship Conclusion**
* **Evidence / Graph**
* **Unit tests result**
* **Task**

(Part 1)

1. You are to implement three (3) methods (*repeat*, *getClock*, and *toMillisecs*) of a class called *Timer*.
2. check the implementation by running the unit tests in *BenchmarkTest*and*TimerTest*.

(Part 2)

1. Implement *InsertionSort*by simply looking up the insertion code used by*Arrays.sort.*
2. run the unit tests in *InsertionSortTest.*

(Part 3)

1. 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.
2. 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.

* **Output screenshot**

**图形用户界面, 文本

描述已自动生成**

* **Relationship Conclusion**

As the graph showed in evidence, the relationship among times of four different arrays for insertion sort is:

***Reverse-ordered > Random > Partially ordered > Order***

* **Evidence / Graph**

**图表, 折线图

描述已自动生成**

**Picture 1- Evidence of Relationship Conclusion**

* **Unit tests result**

(Part 1) unit tests in *BenchmarkTest*and*TimerTest*.

**图形用户界面, 文本

描述已自动生成**

**Picture 2-unit tests in BenchmarkTest**

图形用户界面, 文本

描述已自动生成

**Picture 3-unit tests in TimerTest**

(Part 2) unit tests in *InsertionSortTest*.

图形用户界面, 文本

描述已自动生成

**Picture 4-unit tests in InsertionSortTest**