COMP202 SPRING 2017

BONUS PROJECT: SORTING CONTEST 19.04.2017

***Your submissions will be checked for plagiarism and your grade will be 0.*

***Read the description carefully and understand it well.*

You **are not** allowed to use any java library that substitutes the parts you are asked to implement and solve. Otherwise, your grade will be 0.

1 DESCRIPTION

You were provided with code that can benchmark various sorting algorithms with various lengths and types of data. For this bonus project, you are asked to submit a fast and generic sorting code to be tested against other students' codes. Simply using quicksort or mergesort will not be enough so you have to find faster algorithms online. You will be graded on your performance and correctness. If you submit a correct sorting algorithm, you will receive **1.5%** (worth half a quiz). After we run all the code, we are going to sort the results and divide the results into 3 intervals: Bottom-most 1/3 will not receive any extra credit. Mid 1/3 will receive additional **0.5%** (total 2%), Top 1/3 will receive additional **1.0%**. (total 2.5%). In addition, the top 5% will get **0.5%** more (for a total of 3%). **IMPORTANT**: There will be a time limit on your code. If your implementation fails to terminate correctly until this time limit, you will get a 0. Do not just submit an insertion sort implementation. The faster your code runs the more scrutiny it will receive so make sure you are not utilizing any external sorting library and make sure that you can explain your algorithm.

Grading Summary (Percentage of the final total of the class):
Bottom 1/3: 1.5%
Mid 1/3: 2%
Top 1/3: 2.5%
Top 5%: +0.5% = 3%

2 IMPLEMENTATION

We are going to give you 2 files, to go along the existing sorting code we have provided. One of them is an example class that can do sorting, "Sorting99999" and the other is "TestWithReflection" which is another piece of test code. The sorting class implements insertion sort and the test class shows you how we are going to evaluate your sorting codes.

You are going to submit: "SortingXXXXX.java" where the XXXXX corresponds to your student ID with no leading zeros (i.e. 12345 not 0012345). Do not submit Sorting99999

Things to pay attention:

- Correct name for your class!
- Your class extends *AbstractArraySort* class from the existing sorting code
- Your class implements the default constructor with the correct name
- Your class implements the sort method.
- When you put your class name into the provided test, it runs correctly

RULES

1) You **are not** allowed to use any java library that substitutes the parts you are asked to implement and solve.
2) Your code will be tested with different input files.
3) Your submission will be checked against plagiarism.