CS2401 – Week 5

With this lab assignment, we are going to practice sorting, searching, and discussing performance of algorithms. Searching and sorting are central to many tasks in computer science. It is therefore essential to understand how these work, what to expect from them and how to make the best out of them. We are going to do just this in this lab assignment! Let's get started!

What is the problem?

In class, you have studied algorithms for searching through data and sorting data. You studied linear search and binary search, selection sort, insertion sort, bubble sort, merge sort, and quick sort. You even read about a few other sorting algorithms.
Now, let's wonder together if we could do better with the tools we know (which are the above algorithms).

Your job in this lab assignment is to explore a variation of merge sort or quick sort with a quadratic sorting algorithm (selection, insertion, or bubble), and to identify if this variation can do better than merge sort, quick sort, or your chosen quadratic algorithm.

Variation of sorting algorithms
To do this, you will have to make a hypothesis about the variation you want to try. You will have to justify why you picked this variation and how you expect it to do better than quick sort alone, merge sort alone, and your chosen quadratic algorithm alone. You are expected to follow a scientific approach in which you then conduct experiments to test your hypothesis (you have to describe and justify your experiments), then report on your experiments, and reach a conclusion in which you reflect back on your hypothesis.

So, how will this work? What will you have to do?

In a docx file named YourLastNameYourFirstName-lab4.docx, you will have to:
1/ **propose an algorithm combination** (merge sort or quick sort + a quadratic algorithm). You have to describe the type of combination (e.g., in which order do you combine them, when, why, etc.) – you will have to provide its pseudocode – and to justify your choice of algorithm combination (why do you think your combination will yield better performance than either of the algorithms involved).
2/ **describe the testing strategy** you will put in place to study the performance of your proposed algorithm in comparison to that of merge sort, quick sort, and your chosen quadratic algorithm (see Lab 2 for reference on identifying performance through testing).

Once you have completed steps 1 and 2, you will have to:
3/ **implement** (in a java file called lab4.java) **each of the 4 algorithms** (the quadratic algorithm you chose, merge sort, quick sort, and your proposed algorithm). These algorithms should each be implemented to sort an array of integers.
4/ **carry out your testing strategy** and **report your results** in your file YourLastNameYourFirstName-lab4.docx. Keep in mind that your testing strategy should aim to compare the performance of your proposed algorithm with the performance of merge sort, quick sort, and the quadratic algorithm you chose.
5/ **draw conclusions** and write them in your YourLastNameYourFirstName-lab4.docx file.

Grading:

10 pts   Code is properly indented

 5 pts   Code is well documented

15 pts   Merge sort, quick sort, chosen quadratic algorithm are properly implemented

15 pts   Proposed algorithm is well implemented

15 pts   Well articulated and justified proposed algorithm

15 pts   Well articulated and justified testing strategy

15 pts   Neat and clear report of results and performance comparisons

 5 pts   Clear and justified conclusion

5 extra pts   Propose a follow-up project to further explore the improvement of sorting algorithms

Due date: March 1st at 11:59pm

How to submit?

You have to submit your java file, your word file, and your five test cases in a zipped folder, as a note on piazza, under folder lab4. Your zipped folder should be named YourLastNameYourFirstName-lab4 and your zipped file YourLastNameYourFirstName-lab4.zip.

Failing to follow submission instructions and guidelines will result in up to 15 points off your overall grade in this lab. So please pay attention.

Additionally, your word file is expected to be neat and clear. Failing to do so will result in up to 15 points off. On the other hand, extra neat and clear work will be rewarded by up to 5 extra points.