CS 3410 - Homework 07

Due date: see course Schedule and Blackboard.

Overview

You will compare sorting algorithms.

Directions

Obtain the download, *hw7.zip*. There, you will find *Sort.java* which contains the sorting algorithms: Insert, Shell, Heap, Merge, and Quick. The *TestSort.java* contains a driver to test these sorting algorithms using an array of 1000 integers. You will modify this driver.

Requirements – C Version (75 points maximum)

1. Setup a framework to time each of these sorting algorithms, along with Arrays.sort using 1000 random doubles between 0 and 1000. Do this 5 times for each sorting method and compute the average for each.
2. Make a table that shows your results.

Requirements – B Version (85 points maximum)

1. You will compare: Arrays.sort, Heap Sort, Merge sort, and Quick sort. Similar to C version, generate 1000 random doubles between 0 and 1000. Do this 5 times for each sorting method and compute the average of each. Repeat for 2000 and 3000 random doubles between 0 and 1000.
2. Make a graph of the results.

Requirements – A Version (100 points maximum)

1. You will compare Arrays.sort, Quick sort, and Bucket sort in the following situation. Generate 1000 random integers between 1 and 100. Use a method similar to that in *TestSort.java* where you will put in 10 1's, 10 2's, etc. and then call the *permute* method to randomize the array before sorting.
2. Code Bucket sort so that the number of buckets is a parameter. Run this code for 3 values of the parameter, say, 5, 10, 20 (or any that you find interesting)
3. Replicate all runs 5 times, and take averages.
4. Show a table of your averages.

Deliverables

1. A Word document in this format:
   a. Title page:
      CS 3410 – HW 07, Version (A,B,or C)
      Name
      Date
   b. Answers to question(s).

2. All Code.
3. Email me a zip file, *hw07-lastname.zip* with these items.