CS 2003 Fundamentals of Algorithms and Computer Applications

Experimentally comparing sorting algorithms

In this lab, you will run the six sorting algorithms: Selection Sort (**SS**), Bubble Sort (**BS**), Insertion Sort (**IS**), Merge Sort (**MS**), Quick Sort (**QS**), and Radix Sort (**RS**) on different sized data sets containing natural numbers. You will also evaluate whether the observed differences are statisitically significant using the *paired t-test.* All the sorting algorithms are implemented in `Sorting.java` located in `~class_sandip/2003` (You do not need to modify this class). You can generate the javadoc to obtain the signature and the arguments for each method. You need to create and implement a class `SortDriver` where all sorting algorithms run **on the same data set**. The `main` method must perform the following tasks:

- Generate data sets: use problem sizes 100, 1000, and 10000. Create 10 randomly generated data sets for each problem size. The **same data set** will be sorted by each algorithm. Each data set will contain only integers in the interval $[1, 999]$, and the numbers should be drawn using a uniform distribution over that domain (use the `Random` class).

- Calculate the average and standard deviation of the time taken by each each algorithm for each problem size and store the values in the file `timings.dat`. A line of the file must be formatted as follows: the first token must be the data set size, it is followed by six pairs of tokens, where each pair represents the mean and standard deviation for a given algorithm (use the following order: **SS, BS, IS, MS, QS, RS**). Tokens should be separated by spaces.

Given N data sets on which performances are being evaluated (in this case $N = 10$), the average and the estimated standard deviation of the performance of the algorithm i is given by:

$$\overline{x_i} = \frac{1}{N}\sum_{j=1}^{N} x_{ij} \text{ , and } s_i = \sqrt{\frac{1}{N-1}\sum_{j=1}^{N}(x_{ij} - \overline{x_i})^2}$$

where x_{ij} is the time taken by the algorithm i on the jth data set.

Given algorithms A and B, the t-value for their performance is calculated as

$$t = (\overline{x_A} - \overline{x_B})\sqrt{\frac{N(N-1)}{\sum_{j=1}^{N}((x_{Aj} - \overline{x_A}) - (x_{Bj} - \overline{x_B}))^2}}.$$

The above expression can also be written as $t = \frac{\overline{x_{AB}}\sqrt{N}}{s_{AB}}$, where $\overline{x_{AB}}$ and s_{AB} is respectively the mean and the estimated standard deviation of the performance differences of the two algorithms. (**Note:** When comparing two algorithms, use the algorithm with the higher performance mean as algorithm A.)

Write a report that discusses the average running times of the algorithms on different problem set sizes. Include a graph that plots the average running times and standard deviations as error bars (the class TA can assist with *gnuplot*). In addition, in another set of tables, one for each data set size, show the t values for each pair of algorithms and note if the performances of the two algorithms are statistically significantly different for the corresponding data set size.

If $t > 1.812461$ for any two algorithms for any problem size, then with 95% confidence we can say that the observed performance differences of the two algorithms on the given problem size is statistically significant. In a table Y show the t values and which of the algorithms (use none if applicable) performs statistically better, for all possible combination of algorithms.

Submit the completed class `SortDriver.java` and the report.