Programming Report # 2

Sorting

Due February 15, 2010

For this report, you are going to test three different sorting algorithms: bubble sort, merge sort and Quicksort. For each, there will be implementation aspects as well as testing aspects. Each sort will have special questions to answer and properties to prove. You will produce a report explaining your methods, results and conclusions.

Bubble Sort

Program 6.14 in the text has an implementation of bubble sort. Can this implementation have O(N) performance? Prove your answer visually. That is, produce a graph of run times for many different values of N that proves your point. If is does not produce O(N) performance, your graph should show that, even with optimal data, it still has O(N^2) performance. If it does produce O(N) performance, your graph should show that optimal data does produce O(N) performance.

If your answer is no, implement bubble sort in a way that can produce O(N) performance. Prove that it does visually.

Merge Sort

Implement merge sort. Demonstrate its average case complexity visually. Can merge sort reach O(N) performance? If so, prove it visually. If not, show its best-case complexity.

Quicksort

Implement Quicksort. Show its average-case complexity. Create an algorithm to produce data for various values of N that will cause your Quicksort implementation to demonstrate O(N^2) performance. Show this behavior visually. Hint: the dataset you produce will be tailored to your pivot strategy.

Conclusions

For your conclusions, discuss these issues:

- What are your conclusions about the value of each of these algorithms?
- Which of these three is most viable for heterogeneous data sets?
- Are there any situations in which the other two sorts are viable?