



CALIFORNIA STATE SCIENCE FAIR 2007 PROJECT SUMMARY

Name(s) Camille I. Davis	Project Number J1303
Project Title Awesome Algorithms	
Objectives/Goals To test the efficiency of the three sorting algorithms that I selected, I had to create a random set of data for the algorithms to sort, and write a program to record how long it took for the algorithm to sort the data. One of the problems that I encountered was that some of the algorithms codes that I used could not be compiled because of multiple errors. I also had to find a code that would record how long it took for the algorithm to sort that data.	
Abstract Methods/Materials Computer System: Operating System: Microsoft Windows XP Professional; Model: Dell Latitude C840; Processor: 2.00 GHz Mobile Intel Pentium 4; Memory: 1.0 GB RAM. Programming Language/Compiler: Java SDK 1.5.0_09; Java 2 Runtime Environment, Standard Edition 1.5.0_09-b03; Java VM 1.5.0_09-b03. Editor: TextPad 4.7.3. 1. Obtain Java code examples of the sorting algorithms (BubbleSort, QuickSort, MergeSort, and HeapSort) from the world wide web. The Java code for each algorithm is compiled in a separate program, a program representing each algorithm. 2. Write a program (TestSort) to create the data, run each sort on the data and print the results to a file. 2.1. TestSort creates an array data structure that can store 10,000 records, 50,000 records, and 100,000 records. TestSort is recompiled each time the data structure size was changed. 2.2. The actual data simulated SAT scores that ranged from 1 to 2400. The data is generated using a random number generator (Math.random). 2.3. For each record set (10K, 50K, 100K) each sorting program runs to sort the data. 2.4. The data and the results from the sorting programs are printed out to a file.	
Results Overall, the heapsort algorithm sorted the data the fastest. My experiment was to examine the efficiency of sorting algorithms, which utilized the quicksort, heapsort, and mergesort algorithms. The results of my experiment demonstrated that the heapsort algorithm sorted the data most efficiently. My hypothesis was incorrect. The quicksort algorithm was the slowest algorithm I tested excluding the control.	
Conclusions/Discussion My hypothesis was incorrect. The quicksort algorithm was the slowest algorithm I tested excluding the control.	
Summary Statement My project is about the efficiency of sorting algorithms when it comes to sorting data.	
Help Received My father helped me write the Java programs.	