Lab 7

Goals-
Use simple algorithms for searching
Use simple algorithms for sorting

1. Create data files.

You will need at least four text files filled with random integer values. You don't need a large number of values. Around 20. This will just make testing easier. Create a file with a random collection of the values; 1, 2, 3, 4, 5, 6, 7, 8, & 9. Save it. Make 3 copies. In each copy put in a single value of 0; towards the beginning of one file, around the middle of another and towards the end of the third. Rename each as appropriate, i.e. early, middle, etc. Keep one without a 0.

2. Search for the target value (i.e. 0).

The book or the lecture has several examples of the code for searching algorithms. Implement one of them in a program that searches for the target value in your data files. Compare the results. You cannot use binary search yet. Why? Cite which one you used (page # or slide #).

3. Sort a set of values.

The book or the lecture has several examples of the code for sorting algorithms. Implement one of them in a program. Your program should write the sorted results into a file using a name the user provides. Sort all input files and compare the results. Cite which one you used (page # or slide #).

4. Search for the target value, redux.

Find an algorithm for binary search. Implement it in a program that searches for the target value in your data file. Remember that you cannot use any of the original files. Cite the reference for the algorithm you use. NOTE: Code is NOT an algorithm.

NOTE: You can implement each as a separate program or create a single program giving the user the choice of which activity to perform.

Grading

Programming style- 1 point

Create the 4 necessary input files- 1 points

Implement and test the searching algorithm- 3 points

Implement and test the sorting algorithm - 3 points

Implement and test the binary search algorithm- 2 points