Oberon Ilano

CS 430

October 25, 2019

MinMax Analysis

Analyze the algorithm for readability:

The variable names for the algorithm are good and short. The meaning or the value of the variable names is understandable. The white spacing of the algorithm is great and just right for the readability. The algorithm has good use of indention, it is easy to read and follow the steps. I believe that anyone who understands the algorithmic language can easily follow and understand exactly what the algorithm is attempting to accomplish without being given any further explanations.

Analyze the algorithm for the proper use of algorithm language adopted for class and correctness:

The algorithm does not have an array that begins at 0. It does not use any square brackets for array references, and it does not use semicolons for statement terminator. The algorithm does not have "=" for assignment operator and the use of the same assignment is consistent. It has a correct relational operator and the correctness of the algorithm is very well.

Perform a time complexity analysis

* Define the basic operation of the algorithm:

The basic operation of the algorithm is having two variables, low and high. The variable low stores the minimum element of array S and the variable high stores the maximum element of array S. A loop iterates over each element of array S. If the variable low is greater than S(i) then update variable low with S(i) and if the variable high is less than S(i) then update variable high with S(i). Minimum and maximum elements will be outputted when the iteration reaches the end of the array S.

* Express as a function of
* Find a best approximation for the Big function for

Recurrence Relation:

Implementing a program to test the algorithm:

I did not change anything with the algorithm to make it work in the c program except adding some pointers. The c program does not support pass by reference, in this case, I had to pass the value by using the pointer. The prompt allows the users to choose to create a file with random numbers with a size between 1 and 100, or allow the users to open any text file that contains integers up to 100. If the file is opened, then the findLargestAndSmallest function will be operated to find the minimum and maximum of the array. The algorithm has been tested with different files that contain, ascending, descending, duplicate, or random, numbers.

Overall the algorithm of the function is great, and the author is knowledgeable.