SORTING USING QUICKSORT AND SHELLSORT

REQUIREMENTS AND ASSUMPTIONS:

You will be sorting a set numbers each of which has 10 digits. This will be done using two sorting algorithms: quicksort and shell sort. You will implement both of these algorithms. Then you will compute the average time complexity by practical averaging techniques. Use curve-fitting technique to find the time complexity.

To find the average time complexity run the program and record the time a particular input size. In order to get the average time of a particular input size, you need to generate several data sets of same size and then compute the average of the times. However each data set must contain only 10 digit numbers.

For quicksort the curve is expected to be a multiple of O(nlogn). For shell sort use two different h-sequence that is given in the class. For shell sort, set h-sequence to $(h_2,h_1)= (1.732n^{1/3}, 1)$ then time complexity is expected to be $O(n^{5/3})$. If size of the h-sequence is logn and $h_k = 2^k - 1$ for 1<=k<=logn, the time complexity of shell sort is expected to be $O(n^{3/2})$

INTERNAL REQUIREMENTS:

Use some built in function call for finding the CPU time, or you can get the current time before and after the routine and subtract the two. To compute the averages, use the following technique. Use the data sets of same size and collect the average of those. Then move to the data set of the next size. You may want to use data sets of sizes 10, 20, 30, 40, etc up to 1000. The generation of data should also be a part of the program but not part of the time complexity computation. Once the data is collected, use some curve fitting technique to find the average case time complexity. The curve fitting can be done on any of the machines (including PC). But upload the output file to the nova server.