

# CS1713 ALGORITHM DESIGN AND ANALYSIS

## Assignment 2

### Idea

In this assignment you will do timings for Shell Sort using different h sequences.

### Method

1. Write a Java class with 7 static methods - Insertion Sort and 6 versions of Shell Sort that use different sequences of h for an array of size n. The different sequences are (remember they start with h at the highest value and end with h = 1):

ShellSort1: h = 1, 4, 13, 40, 121.....n/3 (approx.) each h value is 3 \* previous h + 1

ShellSort2: h = odd numbers starting at n/2 (approx.) and ending at 1

ShellSort3: h = every number from n/2 to 1

ShellSort4: h = n/2, n/4, n/8 ending at 1

ShellSort5: h = start at n/2 and take the square root of h to get the next h

ShellSort6: h =  $2^k - 1$  as k decreases to 1. The first value should be the max  $2^k - 1 < n$

2. Write a driver file to do the following things:

- Show that each of the 7 sorts is working on a random int array of size 20. Display the array before and after each sort.
- Print the time taken for each of the 7 sorts on the **same random array of a particular size**. Find a size where the fastest of the sorts be at least 0.2 of a second. Create a table that looks like this:

Size of array =				
Sort	Time1	Time2	Time3	Ave Time

7 rows in the table – one for each sort

- Time the 7 sorts on a random array for sizes N, 2N, 4N, 8N where you will determine N. You will need to use different N starting values for different sorts. Run the code three times and create a table that looks like:

Size	Time1	Time2	Time3	Ave Time

4 lines in table for each sort

### General Notes

1. Timing should be done to 2 or 3 decimal places. When running your timing program first close down all other applications, turn off screensavers, automatic logouts/resets/sleep and take all your measurements during the same session. Allow plenty of time.
2. The choice of N for each sort is very important. It will depend on the speed of your computer and the speed of the sort. N will vary among the sorts so you may need to comment out slower sorts.

### Turning in the Assignment

Upload the tables, source code and compiled code by the start of class on Monday September 22.