Algorithms – Wesley Wong 103055362

1. A Variation of Insertion Sort is Shell Sort.

How is it better than Insertion?

Implement both Insertion and Shell.

Use these implementations to sort the 15,000 numbers contained in unsorted\_numbers.csv.

Insertion Sort has a lot of swaps while being adaptive to the point where after each outer loop a part of the array will no longer need to be sorted.

Shell Sort extends from Insertion Sort, where the original list will be broken down into sublists, each of which will be Insertion Sorted. Hence while Insertion Sorting can only be performed with one sequence, Shell Sorting has a Diminishing Increment when sorting, making it able to sort various elements (both large and small) to the correct part of an array at a reduced timeframe. So while Insertion Sort has more accuracy/stability, Shell Sort can deal with many various data at a faster pace.

1. Code a linear search and binary search.

From the list that you have sorted search for the following 10 numbers using both types of search.



Time/Analyse the searches and compare against the known Big O notation for the Linear and Binary searches.

1. Could Merge Sort be run as a multi-threaded application?

Would there be likely to be a performance gain in doing so?

Why/Why not?

Merge Sort can be run as a multi-threaded application, as it divides an array or list into two halves, merging them together when sufficient depth has been reached.

As threads are individually lightweight processes, multi-threading will enable the user to run threads simultaneously, leading to an improvement in performance time.