The goal of this project is to explore various sorting methods. I will be implementing two different sorts, a shell sort and an improved version of the classic bubble sort. To do this, there are four necessary functions. The first two functionsare not involved in any of the sorting. They will be used to load and format a text file into a usable array, and then save a sorted array into a text file.

The shell sort operates by taking a gap size and separtating the array of data into that many sub arrays. Each array is filled with numbers that are the gap size apart from each other in the main array. After splitting, an intertion sort is performed for each array. The sorted sub arrays are then combined and the process is repeated. A gap sequence is needed to determine how many times to perform this process efficiently while ensuring the array is sorted at the end. This sequence has been provided and will be implemented as part of the shell sort function.

The improved bubble sort will be a modified version of the classic bubble sort. The modification will be the usage of a gap sequence, similar to what was used in the shell sort. This sequence has also been provided and utilizes division, so the gap sizes will need to be floored within the function.