Here is how I planned to write the first project of ECE 36800.

For Load_File, I decide to first use ‘fscanf’ to find out size and assign it to the argument, then allocate memory of an array with size number of pointers and use ‘fread’ to read all the integers into the array and ‘atof’ each element and assign them into an array of ‘long’s and pass it back.

For Save_File, I’ll first ‘fprintf’ the ‘size’ along with a ‘\n’, then use a while loop to put each element of the array into the file followed by a ‘\n’ until reach the end of the array.

For Shell_Insertion_Sort, I’ll start by create the ‘seq1’ by allocate a array of length equals to ‘size’ then assign the first 3 number of the sequence (hand calculated) then rewrite the rest of the algorithm given in the link in the instruction in the loop, use a while loop to generate the sequence and assign them into an array until the newest element is bigger than the size passed in. Then start from the last element of ‘seq1’, use a while loop to sort the function. In each loop, first divide ‘size’ by the current element of ‘seq1’ to find the ‘gap’, and round it down to the nearest integer. Then start from the first element of the array needs to be sorted, each time sort the ‘gap’ number of element using insertion sort(can try bidirectional here), and move the cursor to +’gap’ of its original place until the cursor is beyond size-1, then sort the remaining elements. After the seq1 reach the first element and finish sorting. Return NULL.

For Improved_Bubble_Sort, simply use a loop to find out how many 1.3 times together can just be larger than size and record the rounded number of each of the iterations to create the sequence 2, the rest are basically the same as the above algorithm except use bubble sort instead of insert sort in each iteration of sorting.

For Save_Seq1 and Save_Seq2, simply generate the sequences using the same method mentioned above and save them in the way of Save_File.