mergesort: 15 points

Merge Sort is a sorting algorithm that keeping dividing a list into two halves until it reach a list of 1 element. Then, it merge the individual list back to the original list levels by level from left to right.

The merge function uses similar I, j, k to keep track of the first, mid, and last elements of the list. we need to merged the numbers into a temporary array called mergedNumbers. Then, we have the leftPos and rightPos variables to keep track of the element position of the two lists about to be merged. The first while loop will compare the elements on the left list with the right list to find the smaller element to merge into the temporary array until one of the two list is. The next while loops will be used to copy the rest of the partition array that is leftover from the comparison to the temporary array. The temporary array sorting finish and then copy back to the original array by the for loop.

quicksort: 15 points

Quicksort is another sorting algorithm that also divided the list into two halves using the midpoint as the pivot. The partition function then uses L and h variables to keep track of the elements on the left half and the right half. The first while loop will increment L until L reach an element larger than the pivot. The second while loop will decrement h until h reach an element smaller than the pivot and then, these two element will be swapped to each other. The swapping continues until all elements larger than the pivot is in the right half and all smaller element to the left half or when L = h.

The list is partly sorted with partition function and then recursively calls quicksort to divide each half of the list into smaller halves and sort them with partition function until all the elements find the correct positions.

Testing

All test passed so we can verify that our implementation are correct.

timings: 10 points

Quicksort and Mergesort are much faster compared to other sorts like bubble sort and insertion sort. As the size of the array increase, the time consumption for bubble sort and insertion sort increase much significantly. Merge Sort and Quicksort have better runtime because it keeps dividing the input into halves. Based on the result, we can conclude that Quicksort is the fastest, Merge Sort is the second fastest ignoring the builtin sort.