Assignment 1: Hybrid Sort

Merge sort is known to be one of the fastest sorting algorithm having a running time of O(nlogn). For this project, I decided to use insertion sort to make a hybrid sort because amongst the other in-place sorts we went over, insertion is the easiest to understand and I feel has the best “best case” running time (being equal with bubble sort but I just prefer the logic of insertion).

What I found with this is that even with the running time of insertion sort being O(n2), hybrid sort turns out to be faster than merge sort (depending on the runsize value)! It is because for a certain size, different sorting algorithms could be faster than merge sort. So, combining both sorts would really optimize the running time with the perfect runsize value!