

# Algorithms Assignment 2

Maria Molloy

September 30, 2020

Sort Type	Comparisons	Asymptotic Runtime
Selection Sort	221445	$O(n^2)$
Insertion Sort	114015	$O(n^2)$
Merge Sort	3598	$O(n \log_2 n)$
Quick Sort	$\approx 6500$	$O(n \log_2 n)$

## 1 Selection Sort

Selection sort has an asymptotic runtime of  $O(n^2)$  because it is a nested loop. Therefore it must run through the  $n$ -length array  $n*n$  times. Each time it runs through the first loop, it runs through the second loop  $n$  times. Selection sort has the same asymptotic running time no matter the order of elements in the array.

## 2 Insertion Sort

Insertion sort has a worst case scenario asymptotic running time of  $O(n^2)$ . This happens if the array is in the opposite order of the sorted array. If this is the case, then insertion sort works similarly to selection sort. This is because in the worst case scenario, the nested while loop runs  $n$  times.

## 3 Merge Sort

Merge sort has an asymptotic running time of  $O(n \log_2 n)$  because of the recursion used and recursion binary tree. As the array is divided the running time is  $O(\log_2 n)$ . Then, it is built back up which takes the same time as the length of the array, or  $n$ . This means that adding the run time of the divide aspect and the conquer aspect together, we have a total asymptotic run time of  $O(n \log_2 n)$ .

## 4 Quick Sort

With randomized Quick Sort, the asymptotic run time is  $O(n \log_2 n)$ . This is basically for the same reason that merge sort has an asymptotic run time of  $O(n \log_2 n)$ . As the array is divided with recursion, the depth of the recursion tree has asymptotic run time of  $O(\log_2 n)$ . Then  $O(n)$  amount of work is done at each level of the recursion tree. This gives us a total asymptotic run time of  $O(n \log_2 n)$ . It is important to note that by randomizing the partition, the algorithmic runtime is  $O(n \log_2 n)$ . Without randomizing the pivot, the worst case scenario run time for Quick Sort would be  $O(n^2)$ . Furthermore, it is important to note that the number of comparisons is not consistent for randomized Quick Sort since a different pivot is picked every time.