

Kristy Nguyen

Dr. Jimi Andro-Vasko

CS 302-1002: Data Structures

30 March 2021

CS 302 Assignment 6

- 1.) Assuming every hash look up takes $O(1)$ time, the runtime of my algorithm would be $4n = O(n)$. The $4n$ comes from the loop to read in the list of students, the loop to read in the list of assignments, the loop that goes through the list and sorts it with the hashmap, and the loop that outputs the information to the console.
- 2.) If the list of students and assignments were sorted, the runtime of this algorithm would be $3n + n^2 = O(n^2)$ for insertion sort, $3n + n \log(n) = O(n)$ for merge sort, and $3n + n^2 = O(n^2)$ for quick sort with the pivot being the minimum or maximum element each time. To do the assignment with a sorted list would not make the runtime any quicker because it still takes $O(n)$ runtime in order to read in both the list of students and assignments and output the list of students with the number of missing assignments.
- 3.) If the list of students were unsorted, the runtime of this algorithm would be $3n + n^2 = O(n^2)$ for insertion sort, $3n + n \log(n) = O(n)$ for merge sort, and $3n + n^2 = O(n^2)$ for quick sort with the pivot being the minimum or maximum element each time. To do the assignment with an unsorted list would make the runtime the same as a sorted list because it is still required to go through all the students and assignments in each list.