CSE 2100 – Data Structures & Analysis of Algorithms

Lab #5 – Sorting & Sorting

Labs are evaluated along axes of correctness, design, and style, with scores ordinarily computed as 3 × correctness + 2 × design + 1 × style.

Getting Started

Navigate to https://bitbucket.org/ChrisIsKing/cse2100/src/ and pull the starter code located inside of the folder named *Lab 5 Sorting*.

Located in this folder contains 3 files (main.c, Sorting_algorithms.c, Sorting_algorithms, Searching_algorithms.c, Searching_algorithms.h).

You will be operating mostly in **Sorting_algorithms.c & Searching_algorithms.c** but feel free to inspect all files.

If unfamiliar with concepts of Sorting review the powerpoint documents located in the Lab 5 Directory.

Tasks

Using your knowledge of Sorting algorithms taught in class you are required to:

- Implement the function `void insertion_sort(int array[], int size)` in sorting_algorithms.c that accepts an input array & sorts it in ascending order.
- Implement the function `void merge(int array[], int start_1, int end_1, int start_2, int end_2)` in sorting_algorithms.c that implements the merge subroutine executed by merge sort.
- Implement the function `bool binary_search(int array[], int key, int min, int max)` sorted array and searches for an key using binary search.
- Implement the function `bool linear_search(int array[], int key, int size)` that searches an input array looking for a given key.