**LAB0 WRITE UP**

**# Description of code organization:**

* The lab0 code has been organized in two files which is the main.c and Makefile. It generates the executable mysort by running make.

**# Description of every file**

* Makefile compiles the code main.c and generates executable that can be run by typing make.
* main.c contains the main code of the project. This file takes the inputs as - - name and - - alg=merge or - - alg=quick. Name argument prints the full name of the person written the code and alg argument takes options as merge and quick which sort the file according to the option specified.
* It also takes a non-option argument which is the unsortedfile.txt. This file will have multiple lines which each line having integer of size 0 till INT\_MAX, which is 2147483647.
* It also takes -o argument which takes option as the sorted file eg.sortedfile.txt which will be created and contain the numbers in sorted manner for the unsorted file, in this case unsortedfile.txt. The numbers will be sorted and arranged line by line in the sorted file, in this case sortedfile.txt.
* The main.c contains the merge sort algorithm functions, quick sort algorithm function and function to read and write to the file. Reading the file is done by using char \*\*array and converting the array to an integer 1d array. This array will be passed to the mergesort and quicksort functions depending on the request from the user.
* The sorted array will then be written to a new or an existing file line by line.
* Merge sort is an out of place sorting algorithm which needs to create duplicate arrays to execute sorting. The time complexity of merge sort is O(nlogn) and in worst case is O(n^2).
* Quick sort is an in place sorting algorithm which does not need to create duplicate arrays to execute sorting and executes in sorting on the same array. The time complexity of quick sort is O(nlogn) and in worst case is O(n^2).
* The file is read using getline which takes row of the file as row of the array and columns of the file as columns of the array. This read array is converted using strtol to an integer array. The char \*\*array which reads input of from the unsorted file uses dynamic memory allocation and reallocates for file content greater than size of initially defined array.

**# Compilation instructions**

* Eg of how to run the project. First compile the file using **make** which creates an executable of the name **mysort**.

**# Execution instructions**

* Then run the executable using the command by typing

**./mysort - - name unsortedfile.txt -o sortedfile.txt alg=quick**  which sorts the unsorted file using quick sort OR **./mysort - - name unsortedfile.txt -o sortedfile.txt alg=merge** which sorts the unsorted file using merge sort.

**# Extant bugs**

* The input values are less than **INT\_MAX,** hence a value of more than 2147483647 will lead to differences in the actual sorted file run using **sort -n** linux command and the **mysort** executable. Hence, the same should be considered when running the file.