Lab1 – Documentation

FORK-JOIN Model – Merge sort

1. Divided the input array as many parts based on the number of threads
2. Each array part is being sorted by one thread
3. After each thread sorts its part of the array, it waits to sort and merge those individual sorted parts

Bucket sort

1. The number of buckets is chosen equal to the number of threads
2. The input array is divided into parts equal to the number of threads such that each part can work on different parts of the thread
3. The buckets are fixed with ranges based on the maximum value of the input array
4. The threads insert the elements from its part of the array into a bucket based on the input value
5. Multiset is used along with vector such that when the thread inserts elements into the bucket, it will be inherently sorted
6. Merge the elements of the buckets and it gives the sorted input

Description of Code Organization

Sort_algorithm_bucket.c

Contains functions related to bucket sort, thread handler for bucket sort

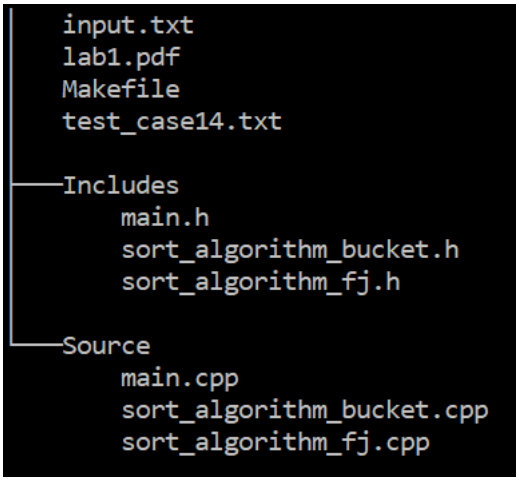
Sort_algorithm_fj.c

Contains functions related to merge sort, thread handler for merge sort

Main.c

Creates threads, parses the command line inputs, forms the input array from the input file, copies the sorted array to the output file

Description of every file submitted



Main.c

Contains the main application

Main.h

Header declarations for the main application

Sort_algorithm_bucket.c

Contains function definitions for bucket sort algorithm

Sort_algorithm_bucket.h

Contains function declarations for bucket sort algorithm

Sort_algorithm_fj.c

Contains function definitions for merge sort algorithm

Sort_algorithm_fj.h

Contains function declarations for merge sort algorithm

Makefile

Helper script to generate and clean the executables

Input.txt, test_case14.txt

Sample input files

Compilation instructions

Make

To generate the executable mysort

Make clean

To clean all object files and executable

Execution instructions

./mysort –name

For displaying the author's name

General rule for application execution:

./mysort [input file] -o [output file] -t [Number of threads] –algs=<bucket or fj>

Eg:

./mysort input.txt -o [output file] -t 10 –algs=fj

Default output stream – standard output (if output file is not provided)
Default number of threads is 1
Input file – mandatory
Algs - mandatory