Aaron Chan
ECE399 Winter2017
Assignment 5

Program Assignment:

This assignment will sort an extremely large array of integers by using a bubble sorting algorithm and implement threading. The following parts are necessary :

(a) sorting of large array with multiple threads (four threads)
(b) sorting of large array with a single thread
(c) timer of both sorting methods to compare the speeds of each.

Upon correct and successful sorting, it will report “Sort Complete”.

Design

Single-Threaded Sorting	
Threading	No additional threads need to be created. The main process itself already counts as one thread, therefore sorting is all that needs to be done.

Multiple-Threaded Sorting	
Threading	To sort this array of integers with four threads, a total of seven threads will be necessary: ● Four to sort quarter partitions of array ● Two to merge-sort the quarter partitions to make halves of array ● Main process itself is a thread, where we will merge-sort to get the full size sorted array
Splitting	We will only need one function to split the array into four sections to start sorting with four threads
Sorting (quarter arrays)	We will initially sort quarter partitions of the original random array with a function that will only sort with a set size. This will implement the bubble sorting code.
Merge-Sort	Merging of pre-sorted arrays will be done and bubble sorting will not be necessary. We will implement a sorting that will mimic a ladder, stepping through each element of the sorted arrays and saving them into a new array in order of lowest to greatest. ● Void function for threads to work with making quarters to halves ● Integer function for main process to finish

Timer

Each sorting method will have their own function that will be called by a main function. Taking advantage of this, we will have the functions return some value of time and display/compare them in main.

Log:

- About 2 hours of reading and designing was done
- About 3 hours was spent to get the code for sorting with multiple threads working properly
- About 1 hour was spent to get timing done and cleaning code up
- About 1 hour was spent to write this document
- An additional 2 hours was spent to implement proper thread merge-sort algorithms and adding a checking function.

Files Turned In

Makefile
header.h
threading.c
sorting.c