Project 2 Report: Application for Threads Sorting
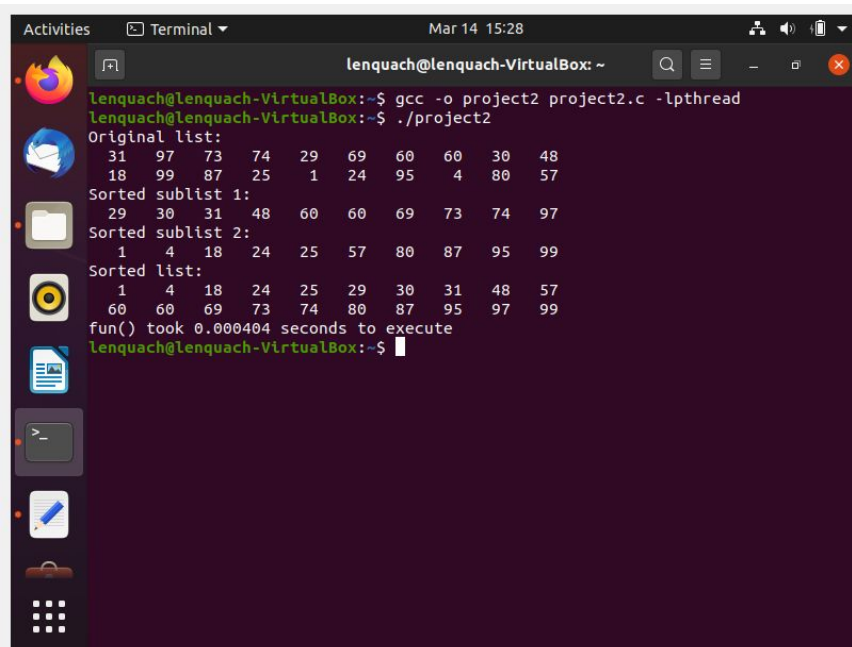## CECS 326 - Operating Systems

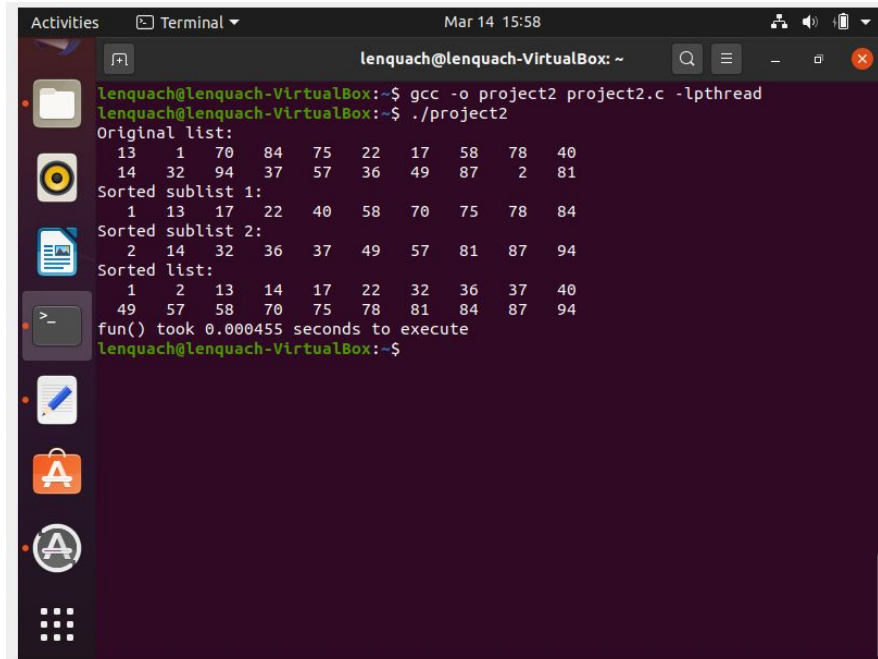**Group members:** Jorel Caoile

Len Quach

Abstract

This project focuses on the topic of threads and the application of using multiple threads. In this project, we have a list of integers that we must sort to increasing order. The list gets separated into two equal lists half the size of the original. Two subthreads are created to sort the two new sublists. A merging thread takes the assorted sublists and assorts them into a single sorted list.

Results

Run 1

Run 2

Discussion

        We have one structure and four functions. The structure is used in creating the threads. The four functions are: *sorter, sort, merge, and sorter_multi. The sorter function takes in parameters which then passes it off to the sorting algorithm. The sort function sorts threads 1 and 2 from the original list. The merge function merges the threads together. The sorter_multi function is the sorting algorithm. We call only the sort function in the main of the program because it uses the sorter_multi which calls the other functions. There is also a clock that outputs how much time the program takes to run. In our code we chose to have N = 20; N being the amount of numbers in the unsorted list. We also chose the largest number of the list to be 99 and for there to be ten numbers per row. We store the numbers in a data array. We use fhalf and lhalf arrays to get the first half and second half of the data array. We sort each of those two arrays and output them on screen. Then we call the sort function one more time to sort the final, merged list.

Individual Contributions

Jorel Caoile:
- Secondary code writer
- Do ReadMe file
- Write report

Len Quach:
- Primary code writer
- Record video
- Report

Video Link: https://youtu.be/9YZwzo6Wmp8