Project 3: Application for Threads Sorting

CECS 326: Operating Systems

Fall 2020

November 7, 2020

Sergio Vasquez 018663448
Kuldeep Gohil 015499534
sergio.vasquez01@student.csulb.edu
Kuldeep.Gohil@student.csulb.edu

Contributions:

Sergio worked on the code, while I troubleshooted it and worked on the report.

Design of the Program:

For the program, our approach was to first create an array of thirteen numbers, which were sorted in a descending order. So the first number in the array was ‘12’ and the last number was ‘0’. Secondly, we created two functions, one was “insertion_sort” and the other was “merge.” The “insertion_sort” function was used to pass a value to the variable ‘a’ that was equal to a structure. Depending on what the variable ‘a’ would equal, the variables ‘lo’ and ‘hi’ were assigned a specific value.
If the variable ‘a’ was equal to ‘0’ for ‘lo’ then it would equal ‘0’.
If the variable ‘a’ was equal to ‘0’ for ‘hi’, then it would equal n divided by 2, where n was equal to the size of the array divided by the size of the position 0.
If the variable ‘a’ was equal to ‘1’ for ‘lo’ then it would equal the value of ‘hi’ when it was ‘0’.
If the structure was equal to ‘1’ for ‘hi’ then it would simply equal ‘n’.
Then we created a for loop, which would continuously run until the value of ‘lo’ plus 1 was greater than the ‘hi’ variable.
There were two functions for the threads. First we created two threads for the insertion sort, which sorted each sublist using sorting algorithm and then we made a merge thread, which would output the content of the global array.

Output for Project 3: