2022 Fall CMPE 322 Project 2
Threading, or not ?

Ömer Faruk Çelik

In this project, you can perform 10 functions given in the description over the random array formed in the input size you will provide.

Execution Details:

After running the make command:
If you write ./main {Array Size} command you can execute the first part of the project which performs functions without creating additional thread while printing execution time to the prompt and result to the "output1.txt".
If you specify thread number (5 or 10) via ./main {Array Size} {Thread Count} you can execute the second part of the project which performs functions over 5 or 10 threads while printing execution time to the prompt and result to the "output2.txt".

Implementation Details:

The main.cpp file contains the functions required to run the desired functions without creating additional thread(first part) and the main function, which is the starting point of the program that reads the program arguments and makes the redirection.
The fiveThread.cpp and tenThread.cpp files run functions over 5 and 10 threads.
In the first part, I defined the desired functions separately and printed the result in the given order.
In the files I wrote for 5 threads and 10 threads, I created functions in the appropriate format to be given to the threads (or modified the existing ones) and I assigned the functions to the threads and made them work properly.
Attention! Mode function returns the smallest when there is multiple mode.

Observations on Execution Time:

When I sorted the array at first, the execution time increased as the number of threads increased, unless the number of inputs was very large, since complexities of min, max and range functions became O(1) and functions such as median, interquartile range took much less time, so maybe the time spent for thread creation can took longer than executing these functions. In order to catch the difference between different thread numbers in terms of execution time, I did the sort not in the beginning, but in the InterquartileRange and median functions.
As a result of my observations, I think that multithreading in this project does not make sense if the input size is not that big.
Some measurement results in the final version of the project:

Array Size	1 Thread Ex. Time(µs)	5 Thread Ex. Time(µs)	10 Thread Ex. Time(µs)
100	488	591	906
1000	890	1015	1102
10000	7625	3233	3149
100000	50368	24810	21253
1000000	258341	121817	106832