Fall 2020

CSCI 6454: HW2 (12 points)

On Shared-memory parallel computing

Problem 1 (5 points): Fibonacci

Write a parallel OpenMP program that computes the N-th Fibonacci number. Use recursive formula for computing Fibonacci numbers. Are you getting your expected results? What challenges did you face and how did you overcome them?

Problem 2 (5 points): Matrix-Matrix Multiplication

Write a parallel OpenMP program to perform matrix-matrix multiplication. You may generate the matrices using random number generators. You may also use any publicly available matrix datasets from the internet. Cite your sources if you use internet datasets. Try to use matrices with dimension greater than or equal to 1000 (row)*1000 (column). Use any approaches of your choice to distribute tasks among threads. If **MatA** and **MatB** are your input matrices, then store the output in matrix ***MatC***. Print C in a file.

Additional requirements: (2 points)

1. For both of the above problems, provide performance evaluation: single-threaded version vs multi-threaded version. Show plots that indicate speedup or parallel efficiency with increasing number of threads. Discuss any insights/findings.
2. Provide at least one set of sample input and output for each of the above problems.

Instructions:

1. For each implementation, follow good programming practices. Have plenty of comments in your source file. Even though I did not quite specify the input and output formats for some problems, you should have descriptive and intelligible prompts and outputs. You MUST have a README file for each problem specifying your input/output and how to run and any assumptions you made.
2. Your programs MUST compile and run. Otherwise, you will lose significant points.
3. For both problems, provide an MS Word document (or similar) with the results, plots, and other discussions.
4. To develop you code and performance experimentation, you may want to use LONI or your own computer. Most modern computers have 4 cores or so. Thus, you should be able to get at least 4 data points for performance plots.

Although, the logical threads can be higher than the number of physical threads.

5. If you want to use LONI, create an account with LONI. Mention my name as your collaborator. I can then have your request approved and share with you my own allocation. Let me know your intention.
6. Put the solution for each problem in a separate directory and zip them together and submit through Moodle. Email me your work ONLY IF Moodle is not working.
7. Name each directory as HW2_YourFirstName_YourLastName_X, where X is the problem number. The final zipped archive name should be HW2_YourFirstName_YourLastName.
8. All work must be your own. This is not a team assignment.
9. DEADLINE: (Soft) 24 September 2020 Thursday 11:59pm.
   Will be accepted without any penalty until 29 September, 2020 Tuesday 11:59pm.