CS4552 Scientific Computing – In18 Semester 8

Department of Computer Science and Engineering
University of Moratuwa

Assignment – False Sharing

False Sharing is one of the hard-to-detect high impactful performance bugs in scientific and high-performance computing. As you learned in the class, false sharing occurs when multiple cores with local caches compete for memory updates in the same shared cacheline.

In this assignment you will write a simple program with false sharing and evaluate the performance impact of it.

Task:

Your task is to write three versions of a scientific program in C language to calculate the value of pi using **Gregory-Leibniz Series (Google it up if you do not know the formula)**.

Version 1: sequential program

Version 2: parallel program with false sharing

Version 3: parallel program without false sharing

You may write your program using either pthreads or OpenMP thread libraries.

- You should parameterize the program execution to change the number of terms of the series you are calculating and for the parallel programs (Version 2 and Version 3), you should have a parameter to vary the number of threads.
- You should run the programs on your personal computer as well as on the server provided previously (let me know if you do not have access) and collect execution time data for all three versions.
- Calculate the value of pi for as many terms of the formula as possible to get a sufficiently large execution time.
- Run the parallel programs for varying number of threads (in powers of two): 1, 2, 4, 8, 16, …… Do this for as many threads as possible which is feasible with the hardware systems.
- Plot the readings on suitable graphs to show performance comparison in an appropriate manner.

Submission: You should submit the source code, any supplementary scripts written for running the code and plotting graphs, and a short report explaining observations and plots as a single zip archive to moodle submission link.

Please refer to moodle for assignment submission deadline.