

# CSC563 Multithreaded Distributed Programming

## Project Assignment 1: Benchmarking

Due on 9/21/2019 11:59PM

### Assignment

This project aims to teach you how to benchmark your computer's computational performance. In this project, you need to design a benchmarking program that measures the CPU speed, in terms of floating point operations per second (Giga FLOPS,  $10^9$  FLOPS) and integer operations per second (Giga IOPS,  $10^9$  IOPS); measure the processor speed at varying levels of concurrency (1 thread, 2 threads, 4 threads, and 8 threads).

You can be creative with this project. You are free to use C, C++, Java, Python and any languages you prefer. You are free to use any machines (PC, UNIX, etc) for your development.

### Requirements

- You must write all benchmarks from scratch. You can use well known benchmarking software to verify your results, but you must implement your own benchmarks. Do not use code you find online, as you will get 0 credits for this assignment.
- All of the benchmarks will have to evaluate concurrency performance; concurrency can be achieved using threads. Be aware of the thread synchronizing issues to avoid inconsistency or deadlock in your system.
- Experiments should be done in such a way that they take multiple seconds to minutes to run, in order to amortize any startup costs of the experiments.
- Not all timing functions have the same accuracy; you must find one that has at least 1ms accuracy or better.
- Since there are many experiments to run, find ways (e.g. scripts) to automate the performance evaluation.
- All benchmarks must be run 3 times, and the reported values should be the average and standard deviation.
- No GUIs are required. Simple command line interfaces are fine.

### Deliverables

You need to submit an IEEE format report on BlackBoard

You need to submit all the source code as a zip file on BlackBoard

In your report, you need to include the following sections:

1. **Design:** describing the overall program design, and design tradeoffs considered and made. Also describe possible improvements and extensions to your program (and sketch how they might be made).

2. **Manual:** A detailed manual describing how the program works. The manual should be able to instruct users other than the developer to run the program step by step. The manual should contain example commands to invoke each of the four benchmarks.
3. **Performance:** Since this is an assignment aimed at teaching you about benchmarking, this is one of the most important part; you must evaluate the benchmarks with the entire parameters space mentioned in Section 1. You must produce graphs to showcase the results. Please combine data and plot on the same graph wherever possible, for either compactness reasons, or comparison reasons. Don't forget to plot the average and standard deviation, as opposed to just a simple value. Also, you need to explain each graph's results in words. Hint: graphs with no axis labels, legends, well defined units, and lines that all look the same, are likely very hard to read and understand graphs. You will be penalized if your graphs are not clear to understand.

## Grading

- Report format (10%)
- Design (10%)
- Manual (10%)
- Performance (50%)
- Source Code (20%)