CMPSC 140 Summer 2022: Initial Final Project Outline

Instructor: Richard Boone

Due dates:
Initial project outline: Friday, August 5th
Initial Project Presentation: Tuesday, August 9th
Final report and presentation: Tuesday August 23rd

Initial Project Outline (5 points)
In a 1-2 paragraph overview of what project you plan to do, and what programming frameworks you plan to use.

I will be reviewing project outlines over the weekend, and will give feedback to everyone.

Project Goals Presentation (10 points)
In 2-3 slides, present to the class what your project is about. Describe the problem you're working on, and discuss overall goals including what frameworks you will use and a rough timeline. There will be a small time period after your presentation for questions from myself and other class members.

Final Project Presentation (35 points)
In a 10-20 minute presentation, describe the work you did. You should include a description of the problem, details about problem scale, speedup, efficiency, and the total number of threads/cores you used. Graphs and tables from your report should be included if relevant to the presentation. Additionally, you should include a detailed description of your method of solving the problem, and the steps you took to get there.

At the end of your presentation will be a brief time for questions from myself or from the class.

Final Project Report (40 points) Your final project report should be a comprehensive overview of the final project on which you worked. You should include a number of sections as follows:

1. Introduction: Describe the problem, as well as any previous solutions of the problem we've discussed in class. Discuss on a general level what you plan to do with your project

2. Methods: Describe in detail the methods you used to program your final project. This should include relevant code sections, programming struc-

tures, a rough timeline of your project development, and the individual responsibilitiers of your team members

3. Results: Here you should include graphs of your runtimes, problem sizes, efficiencies, and any other graphs or data you measured that are relevant to the problem. You should run your problems for large scales, up to 15 minutes if possible for your problem, to measure more accurately the effects of parallelization.

4. Conclusions: Discuss the problem and the way you solved it in a more general sense. Show any discoveries you may have made along the way and explain your results.

Included with your project report should be all of your code and a detailed description of how to compile and run any code you have submitted.

Project topic recommendations:

1. Run a comparison of the three parallel options on huge sizes of standard problems (MPI, pthreads/OMP, MPI). Potential options are as follows:

    (a) The Heat Equation

    (b) The Travelling Salesman Problem

    (c) Matrix-Matrix multiplication using Block computations

    (d) Ising Machine

    (e) Other problems: propose your own.

2. Networked Parallel computing
   Using multiple computers, run a parallel algorithm by passing messages through networking protocols. You may choose your own networking frameworks and protocols, though I recommend using C++ and OpenMP for ease of use. Some networking experience is recommended. You should implement one of the above algorithms or one of the algorithms we've discussed in class. Limit to 2 or 4 computers depending on the difficulty of the algorithm.

3. CUDA:
   Learn and implement one of the algorithms we've discussed in CUDA. You will need access to a GPU of your own, but CUDA will be discussed in class in the later weeks.

4. Suggest a project:
   If there's a project you're interested in undertaking, feel free to propose it. As long as the project is within the realm of parallel computing, and reasonable difficulty within the time limits, it should be accepted

If you're unsure of what to choose, I recommend sticking with option 1. This is the standard expectation and well within what you have already learned

within the class. That being said, I encourage you to find a project that you're interested in, regardless of expected difficulty. Difficult projects will be given more leniency in terms of grading.

Grading: Grading will be done based on four metrics: Project Difficulty, Completion level, Novelty and exploration of the problem, Quality of presentation/report.

https://powcoder.com

Assignment Project Exam Help

Assignment Project Exam Help

Add WeChat powcoder

https://powcoder.com

Add WeChat powcoder