EE-P 524 A
Applied High-Performance GPU Computing

Time: one weekly live online Zoom lecture, xxx from 6:00-8:30 PM

one weekly additional supplementary lecture pre-recorded (Panopto)

Instructor: Dr. Colin Reinhardt (colinrei@uw.edu)

Office Hours: TBD (will take poll of class preferences)

TA: tbd

Office Hours: twice weekly, tbd

Course Overview:

There will be 10 lectures given via Zoom and supplemented with additional pre-recorded video lectures on technical/programming topics. Hand-on programming exercises will be part of the weekly lecture. Also there will be 5 homework assignments which will consist of (a) readings, (b) coding (c) theoretical/mathematical problems. The remainder of the quarter will be focused on a final project which will be comprised of a fairly significant GPU code design and implementation on a topic approved by the course instructor, utilizing and applying techniques learned in the class.

Grading:

- Homeworks (5) 50%
- Final Project 50%

Course Materials:

There is no required textbook for this course.

Reading materials for the course as well as a list of supplementary reading materials will be posted on the course website. Nearly all materials will be available in online electronic formats, either freely available public literature or through UW Library (www.lib.washington.edu)

Prerequisites

- Proficiency programming with C and/or C++ programming and using integrated software development environments (Visual Studio, Eclipse) for building and debugging.
    - The standard template library (STL) will be used.
- Familiarity with vector calculus and partial differential equations (PDEs); physical foundations and formulation of PDEs, Maxwell's Eqns, the wave equation, and the dispersion equation.
- Comfort with applied matrix analysis and linear algebra and numerical analysis, eigensystems, eigenvalue problems and solutions, particularly the basic vector and matrix operations.

Course Policies

You may collaborate and discuss homework assignments and project design and implementation with your fellow classmates, professor, TA and others. However, the work you submit must be your own, and you must write your own code(s). Copying code and plagiarizing is not allowed.

Course Schedule: Lecture Topics & Assignments

Week 1 : Introduction, evolution, and overview of parallel computing. Introduction and overview to OpenCL and CUDA GPU APIs. Basic GPU microprocessor architecture.

Week 2 : Fundamentals of CUDA host and kernel design and programming. Basic GPU kernel analysis techniques. GPU microprocessor architecture.

Week 3 : CUDA host and kernel programming details. Kernel analysis, profiling, debugging.

Week 4 : Parallel software and performance theory. Parallel patterns & algorithms, part 1. Implementation using CUDA.

Week 5 : Parallel patterns & algorithms, part 2. Implementation using CUDA.

Week 6 : Parallel FFT and convolution on GPU and applications in image processing

Week 7 : Extended case study on parallelizing machine learning with GPUs and OpenCL: deep convolutional neural networks (CNNs) and GPU-parallelized training algorithms

Week 8 : Extended case study on parallelizing machine learning with GPUs and OpenCL: deep convolutional neural networks (CNNs) and GPU-parallelized training algorithms.

Week 9 : Extended Case study in physics simulation with parallelized PDEs and efficient interactive 3D scientific visualization with GPUs and CUDA.

Week 10 : Extended Case study in physics simulation with parallelized PDEs and efficient interactive 3D scientific visualization with GPUs and CUDA.