CPSC/ECE 4780/6780
General-Purpose Computation on Graphical Processing Units (GPGPU)

Class Location/Time: Online (Asynchronously with pre-recorded lectures posted on Canvas) Tue, Thu 9:30am-10:45am, Aug 19th-Dec 11th.

Instructor: Dr. Shuangshuang Jin **Email: jin6@clemson.edu** **Office: ZGEC 311 (Charleston)**
Phone: (206) 601-7654 **Office Hours: Mon 10:30am-11:30am**
Webpage: http://jin6.people.clemson.edu

Teaching Assistant/Grader: Liwei Wang
Email: liweiw@clemson.edu **Office Hours: Wed 2:00pm-3:00pm**
Phone: 864-633-7758

Course Description

Graphics processing units (GPU) is a term introduced by NVIDIA in the late 1990s which typically handles computation only for computer graphics. After 2001, with the advent of both programmable shaders and floating-point support on graphics processors, general-purpose computing on GPUs became practical and popular for scientific computing applications with its increasing speed and volume of computation.

Nevertheless, extracting full performance from a GPU is challenging. Parallel algorithms are necessary but far from sufficient. Careful layout of both control flow patterns and memory access patterns is required to avoid flow divergence and bank conflicts, which can severely stall computational threads. Memory hierarchies, memory staging techniques, and the available synchronization primitives must be thoroughly understood to provide tremendous performance improvements over conventional programming techniques on GPUs.

This course is designed to provide instruction in the design and implementation of GPU-based solutions to computationally intensive problems from a variety of disciplines. NVIDIA's CUDA and OpenCL will both be used as the programming language, and inter-operate with the open standard graphics language, OpenGL, for massive data visualization.

Prerequisites: Working knowledge of C programming, linear algebra, matrix manipulation, data structures and algorithms, and Linux system. Knowledge of computer architecture, parallel computing, and computer graphics is a plus.

Course Objectives

At the completion of the course, students should be able to

- Design and implement GPU-based solutions to computationally intensive problems from science and engineering disciplines:
    - Develop computational thinking and parallel programming skills.
    - Implement CUDA, OpenCL, OpenGL applications on Palmetto Clusters.
    - Achieve improved application performance and scalability through understanding of architecture and software mapping.

Required Materials

- **CUDA**:
    - Programming Massively Parallel Processors A Hands-on Approach, David B. Kirk, Wen-mai W. Hwu
- **OpenCL**:
    - Heterogeneous Computing with OpenCL, Benedict R. Gaster, Lee Howes, David R. Kaeli, Perhaad Mistry, Dana Schaa
- **OpenGL**:
    - OpenGL A Primer, Edward Angel

Topical Outline

- Parallel Computing and Palmetto Cluster
- GPGPU
- CUDA Overview
- CUDA Threads
- CUDA Memories
- Parallel Reduction
- Atomics
- Streams
- Multi-GPU Programming
- Case Study: MRI
- **Exam 1 on CUDA Essentials**
- OpenCL Overview
- Programming Details
- Kernels, Memories, Synchronization Events
- CUDA and OpenCL by Comparison
- OpenGL Overview
- Transformations and Projections
- Lights and Materials
- Bitmaps, Images, and Pixels
- Textures
- **Exam 2 on OpenCL and OpenGL Essentials**
- Graphics Interoperability
- CUDA-OpenGL Interoperation

- OpenCL-OpenGL Interoperation
- **Final Project**

Grading

- **Grade Scale:**
  - **Undergraduate Students**:
    - A – 90 to 100;
    - B – 80 to < 90;
    - C – 70 to < 80;
    - D – 60 to < 70
    - F – < 60
  - **Graduate Students:**
    - A – 92 to 100;
    - B – 82 to < 92;
    - C – 70 to < 82;
    - F – < 70

- **Grade Components:**
  - **Assignments:** 50%
  - **Final Project**: 20%
  - **Exams**: 30%
    - **Exam 1:** 20%
    - **Exam 2:** 10%

Additional Policies

- **Attendance**: Attendance is critical to the success of students in the class.
- **Academic Integrity**: Cheating or plagiarizing on any work for this course will receive no credit for that work. Further action will also be taken if necessary.
- **Late-work**: An assignment submitted within 0 to 24 hours after the due time will only be eligible for 80% of the maximum number of point allotted; An assignment submitted within 24 to 48 hours after the due time will only be eligible for 50% of the maximum number of point allotted; Assignments submitted more than 48 hours after the due time will not be accepted.
- **Re-grade**: All requests for re-grades must be submitted within one week of the graded assignments being returned.
- **Disability Accommodations**: If you have a documented disability that requires an accommodation, please contact me so we can set up an appointment to discuss your needs. Or contact: Student Disability Services, G20 Redfern, 864-656-6848.