GPU Accelerated Computing

CS 432 L1

Spring Semester 2022

"Today is just the beginning of Kepler. Because of its super energy-efficient architecture, we will extend GPUs into datacenters, to super thin notebooks, to superphones."

Jensen Huang, CEO @ NVIDIA Inc.

Course Information

Class Location: W-110
Class Meeting Time(s): Wed Fri (11:30 AM-12:45 PM)
Course Prerequisites: CS 232,MATH 205
Hardware/Software Prerequisites: Laptop/Desktop PC, headphones with mic, Internet Connectivity (support live stream of video), Gmail account with access to Google Collaboratory, CUDA Toolkit, NVIDIA Developer Account
Content Area: This course meets the requirements for CS Elective.

Instructor Information

Instructor: Muhammad Mobeen Movania
Title: Assistant Professor
Office Location: Room C-215 Faculty Pod C-203
Email: mobeen.movania@sse.habib.edu.pk
Office Hours: Wed: 9:00AM – 11:00AM
Additional Information
Please book an appointment via email.

Course Description

Modern scientific computing focuses on developing applications using GPUs because of the rapid growth in computing power, and drop in the price of massively parallel accelerators. The abundant data parallelism available in many-core GPUs has been a key interest to improve accuracy in scientific and engineering simulation. Modern GPUs use multiple streaming multiprocessors (SMs) with potentially thousands of cores, fast context switching, and high memory bandwidth to tolerate ever-increasing latencies to main memory by overlapping long-latency loads in stalled threads with useful computation in other threads.

The Compute Unified Device Architecture (CUDA) is a simple C-like interface proposed for programming NVIDIA GPUs. However, porting applications to CUDA remains a challenge to average programmers. CUDA places on the programmer the burden of packaging GPU code in separate functions, of explicitly managing data transfer between the host and GPU memories, and of manually optimizing the utilization of the GPU memory. Hence, the pre-condition in efficient utilization of the GPU resources is a comprehensive understanding of the underlying architecture, and the exertion of complex kernel optimizations.

Course Aims

Learn how to program heterogeneous parallel computing systems and achieve high performance and energy-efficiency, functionality and maintainability, scalability across 2 future generations, parallel programming API, tools and techniques, principles and patterns of parallel algorithms, processor architecture features and constraints. This is a hands-on programming course. We will be using Google Collaboratory environment for programming in CUDA C and pyCUDA, which is free to use with one NVIDIA GPU access. Therefore, you do not need to have personal GPU machine to take this course.

This course teaches the fundamental tools and techniques for accelerating C/C++ applications to run on massively parallel GPUs with CUDA®. Students will learn how to write code, configure code parallelization with CUDA, optimize memory migration between the CPU and GPU accelerator, and implement the workflow that they have learned on a new task— accelerating a fully functional, but CPU-only, particle simulator for observable massive performance gains.

Course Learning Outcomes (CLOs)

CLO 1 - Write code to be executed by a GPU accelerator. Expose and express data and instruction-level parallelism in C/C++ applications using CUDA.

CLO 2 - Utilize CUDA-managed memory and optimize memory migration using asynchronous prefetching.

CLO 3 - Utilize concurrent streams for instruction-level parallelism.

CLO 4 - Write GPU-accelerated CUDA C/C++ applications, or refactor existing CPU-only applications, using a profile driven approach.

Mode of Instruction

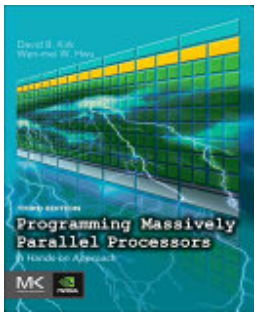
a. **Type of Instruction:** Two Synchronous lecture (75 minutes duration) 100% of course engagement - will be held each week during class timings (online or in-person depending on COVID-19 situation with regards to campus access by the students). Around 100% course engagement will be held through this synchronous mode of instruction. The live sessions will be used for Q&A, group discussions, problem solving, code demonstrations (pair programming) etc. Attendance is compulsory for these classes and students are expected to participate mandatorily both through audio and text during the online mode. Both attendance and participation during these live sessions will be used to gauge the student participation in the course.
b. **Course Participation:** Various instruments would be factored in course engagement. Participation in Q&A, polls, exercises during the synchronous sessions, contribution in creation of digital presence will all be factored in it.

Engagement, Net-etiquettes and Participation Rules

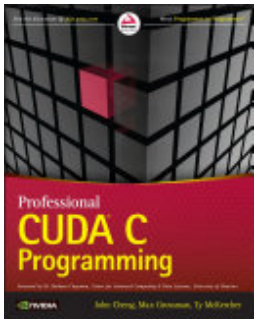
a. For all synchronous lectures students need to keep their microphone and camera ready to be used. Students will be required to keep their microphone and camera on during Q&A part of the lecture.
b. Further Do’s and Don’ts can be discussed in the first two weeks of classes.

Required Texts and Materials

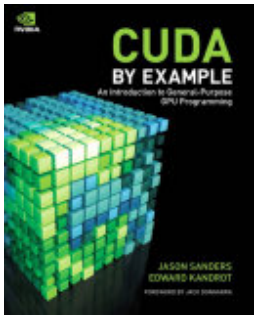
Programming Massively Parallel Processors
ISBN: 9780128119860
Authors: David B. Kirk, Wen-mei W. Hwu
Publisher: Morgan Kaufmann
Publication Date: 2016-11-15



Optional Materials



Professional CUDA C Programming
ISBN: 9781118739327
Authors: John Cheng, Max Grossman, Ty McKercher
Publisher: John Wiley & Sons
Publication Date: 2014-09-09



CUDA by Example
ISBN: 9780132180139
Authors: Jason Sanders, Edward Kandrot
Publisher: Addison-Wesley Professional
Publication Date: 2010-07-19

Assessments

Assessment Component	Percentage
Assignments (3 – 6%+7%+7%)	20%
Project (1)*	25%
Quizzes (5 – 2% each)	10%
Mid Term Exam (1)	20%
Research Paper/Book Chapter Writeup (1)**	25%

Grading Scale

Letter Grade	GPA Points	Percentage
A+	4.00	[95-100]
A	4.00	[90-95)
A-	3.67	[85-90)
B+	3.33	[80-85)
B	3.00	[75-80)
B-	2.67	[70-75)
C+	2.33	[67-70)
C	2.00	[63-67)
C-	1.67	[60-63)
F	0.00	[0, 60]

Note: [a, b) is a range of numbers from a to b where a is included in the range and b is not.

Late Submission Policy

- Students are required to adhere to the timelines given by the instructor and make timely submissions.
- If a student is facing some genuine problem that hinders their performance in the assessment, he/she should discuss with the instructor in advance to get some extension or accommodation. No last-minute requests for relaxation will be entertained.
- Late submissions without any genuine reason will be subjected to a penalty of 10% grade reduction for every delay of 24 hours.

Week-Wise Schedule (Tentative)

Spring 2022 Weekly Schedule*

Week	Description	Readings	Assessments
Week - 1 January 10 – 14	Introduction to heterogeneous parallel computing Add / Drop period	Chapter 1	

Week	Description	Readings	Assessments
Week - 2 January 17 – 21	CUDA programming and execution model, data parallel computing and scalable parallel execution Last day to Drop Course(s): **January 19, 2022** Last day to Add Course(s): **January 21, 2022**	Chapter 2/3	Quiz 1 Assignment 1 (release)
Week - 3 January 24 – 28	Global memory and data locality	Chapter 4	Assignment 1 (due)
Week - 4 January 31 – February 4	Performance considerations, shared memory, constant memory	Chapter 5	Quiz 2
Week - 5 February 7 – 11	Numerical considerations	Chapter 6	
Week - 6 February 14 – 18	Parallel patterns: convolution	Chapter 7	Assignment 2 (release)
Week - 7 February 21 – 25	Parallel patterns: prefix sum	Chapter 8	Assignment 2 (due) Quiz 3
Week - 8 February 28 – March 4	Parallel patterns: parallel histogram	Chapter 9	Project Proposals (due)
Week - 9 March 7 – 11	Parallel patterns: sparse matrix	Chapter 10	
March 12 (Working Sat.)	Mid-term Exam		
Week - 10 March 14 – 18	Parallel patterns: merge sort	Chapter 11	Assignment 3 (release) Quiz 4
March 21 – 25	Conference Week‡ Pakistan Day: March 23, 2022		
Week - 11 March 28 – April 1	CUDA dynamic parallelism	Chapter 13	Assignment 3 (due)
Week - 12 April 4 – 8	CUDA Streams and Concurrency 1st Ramzan†: **April 3, 2022** Last Day to Withdraw from Course(s): **April 8, 2022**	Chapter 6*	Project Interim Demo

Week	Description	Readings	Assessments
Week - 13 April 11 – 15	GPU-Accelerated CUDA Libraries, OpenACC	Chapter 8*/9*	Quiz 5
Week - 14 April 18 – 22	CUDA programming in Python (pyCUDA /numba)	Online Material	
Week - 15 April 25 – 29	Multi-GPU Programming		
April 30 – May 6	Project Demos and Viva Reading Days: **April 30, 2022 – May 2, 2022** Eid ul Fitr†: May 3 – 6, 2022		
May 9 – 14 & 16 – 18, 2022	End Term Examinations Days§		

* These are chapters from Professional CUDA C Programming

Notes:

* The University reserves the right to correct typographical errors or to adjust the Academic Calendar at any time it deems necessary.

† Subject to the sighting of the new moon.

‡ No Class(es).

Attendance Policy

During COVID -19, the existing attendance policy is suspended and replaced with this special policy on student engagement listed on this link (To be provided later on). Expectations - Students are expected to watch all pre-recorded sessions and attend all synchronous sessions. Faculty members will measure attendance in dynamic ways including in class participation, feedback on recorded sessions, performance in assessments etc. Students failing to join any live session must inform their instructor within 24 hours along with the reason. If a student can’t attend any or majority of the live sessions and the nature of the class requires in-class participation then the student can be dropped from the course. Please refer to the COVID-19 attendance policy for more details.

Final Exam Policy

There will be no final exam in this course.

Academic Integrity

Each student in this course is expected to abide by the Habib University Student Honor Code of Academic Integrity. Any work submitted by a student in this course for academic credit will be the student's own work.

Scholastic dishonesty shall be considered a serious violation of these rules and regulations and is subject to strict disciplinary action as prescribed by Habib University regulations and policies. Scholastic dishonesty includes, but is not limited to, cheating on exams, plagiarism on assignments, and collusion.

a. Plagiarism: Plagiarism is the act of taking the work created by another person or entity and presenting it as one's own for the purpose of personal gain or of obtaining academic credit. As per University policy, plagiarism includes the submission of or incorporation of the work of others without acknowledging its provenance or giving due credit according to established academic practices. This includes the submission of material that has been appropriated,// bought, received as a gift, downloaded, or obtained by any other means. Students must not, unless they have been granted permission from all faculty members concerned, submit the same assignment or project for academic credit for different courses.
b. Cheating: The term cheating shall refer to the use of or obtaining of unauthorized information in order to obtain personal benefit or academic credit.
c. Collusion: Collusion is the act of providing unauthorized assistance to one or more person or of not taking the appropriate precautions against doing so.

All violations of academic integrity will also be immediately reported to the Student Conduct Office.

You are encouraged to study together and to discuss information and concepts covered in lecture and the sections with other students. You can give "consulting" help to or receive "consulting" help from such students. However, this permissible cooperation should never involve one student having possession of a copy of all or part of work done by someone else, in the form of an e-mail, an e-mail attachment file, a diskette, or a hard copy.

Should copying occur, the student who copied work from another student and the student who gave material to be copied will both be in violation of the Student Code of Conduct.

During examinations, you must do your own work. Talking or discussion is not permitted during the examinations, nor may you compare papers, copy from others, or collaborate in any way. Any collaborative behavior during the examinations will result in failure of the exam, and may lead to failure of the course and University disciplinary action.

Penalty for violation of this Code can also be extended to include failure of the course and University disciplinary action.

Program Learning Outcomes (For Administrative Review)

Upon graduation, students will have the following abilities:

- PLO 3: Programming: Program a given solution in a variety of programming languages belonging to different paradigm.
- PLO 5: Tools: Work with the latest tools that support development, e.g., IDE's, version control systems, debuggers, profilers, and continuous build systems.
- PLO 6: Self-learning: Research, learn, and apply requirements needed to implement a solution for a given high level problem description.

Program Learning Outcomes (PLOs) mapped to Course Learning Outcomes (CLOs)				
	CLOs of the course are designed to cater following PLOs: **PLO 3:** Programming **PLO 5:** Tool **PLO 6:** Self-Learning			
	Distribution of CLO weightages for each PLO			
	CLO 1	**CLO 2**	**CLO 3**	**CLO 4**
PLO 3	50%	50%		
PLO 5			100%	
PLO 6				100%

Mapping of Assessments to CLOs

Assignments	**CLO #01**	**CLO #02**	**CLO #03**	**CLO #04**
Assignment 1	X			
Assignment 2		X		
Assignment 3			X	
Quiz 1	X			
Quiz 2	X			
Quiz 3		X		
Quiz 4		X	X	
Quiz 5			X	X
Mid Term	X	X		
Research Paper/Book Chapter	X	X	X	X

Project	X	X	X	

Recording Policy

As per HU's teaching policy during Covid-19, all synchronous and synchronous sessions will be recorded and uploaded on our Video Management System (Panopto). Link to the folder of recordings will be available to all students.

Accommodations for Students with Disabilities

In compliance with the Habib University policy and equal access laws, I am available to discuss appropriate academic accommodations that may be required for student with disabilities. Requests for academic accommodations are to be made during the first two weeks of the semester, except for unusual circumstances, so arrangements can be made. Students are encouraged to register with the Office of Academic Performance to verify their eligibility for appropriate accommodations.

Inclusivity Statement

We understand that our members represent a rich variety of backgrounds and perspectives. Habib University is committed to providing an atmosphere for learning that respects diversity. While working together to build this community we ask all members to:

- share their unique experiences, values and beliefs
- be open to the views of others
- honor the uniqueness of their colleagues
- appreciate the opportunity that we have to learn from each other in this community
- value each other's opinions and communicate in a respectful manner
- keep confidential discussions that the community has of a personal (or professional) nature
- use this opportunity together to discuss ways in which we can create an inclusive environment in this course and across the Habib community

Office Hours Policy

Every student enrolled in this course must meet individually with the course instructor during course office hours at least once during the semester. The first meeting should happen within the first five weeks of the semester but must occur before midterms. Any student who does not meet with the instructor may face a grade reduction or other penalties at the discretion of the instructor and will have an academic hold placed by the Registrar’s Office.