CS599: Programming Massively Parallel Processors & Hetrogenous Systems

Understanding and programming the devices powering AI
Lecture 0: Class Overview and Introductions

Jonathan Appavoo **email: jappavoo @ b u . e d u** **office: CDS 706**

CS599: Programming Massively Parallel Processors & Hetrogenous Systems

Understanding and programming the devices powering AI
Lecture 0: Class Overview and Introductions

CUDA Programming on GPU's

Jonathan Appavoo email: jappavoo @ b u . e d u office: CDS 706

CS599

Course Synopsis
(See Syllabus Piazza for details)

Course Prerequisites and Structure

- Prerequisites
  - Basic understanding of the C programming language (especially its use of pointers)
  - Undergrad level knowledge of computer architecture
- Structure
  - Lectures twice a week for ~6-8 weeks
  - In class project Q&A, hacking and discussions remainder of semester
  - Office hours as needed

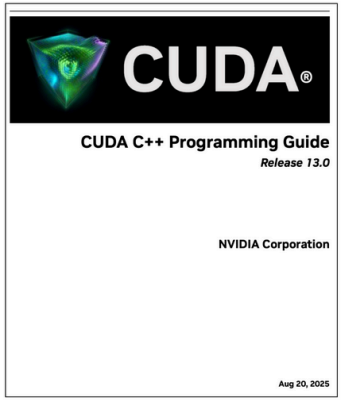
Course Components

- 12-16 Lectures
- Programming Assignments: 2
- Paper readings: ~5
- Midterm exam
- **Group Project**

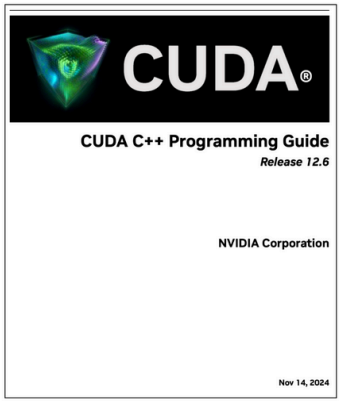
Course Resources

- Primary source of information (besides slides): NVIDIA Inc, "CUDA C++ Programming Guide" (and friends: best practices, API references, white papers, etc)
- Textbook: none
  - but if you need one Kirk and Hwu: "Programming Massively Parallel Processors: A Hands-on Approach (4th edition)" is reasonable.
- Piazza
- Github Classroom
- Gradescope
- MOC/NERC/Red Hat Open AI
  - GPUS: **V100**, A100 and H100
  - CUDA: 12.6.3

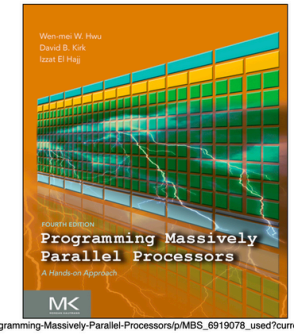
https://docs.nvidia.com/cuda/#programming-guides



https://docs.nvidia.com/cuda/pdf/CUDA_C_Programming_Guide.pdf



https://docs.nvidia.com/cuda/archive/12.6.3/pdf/CUDA_C_Programming_Guide.pdf



https://bu.bncollege.com/c/Programming-Massively-Parallel-Processors/p/MBS_6919078_used?currentCampus=8529&rental=true

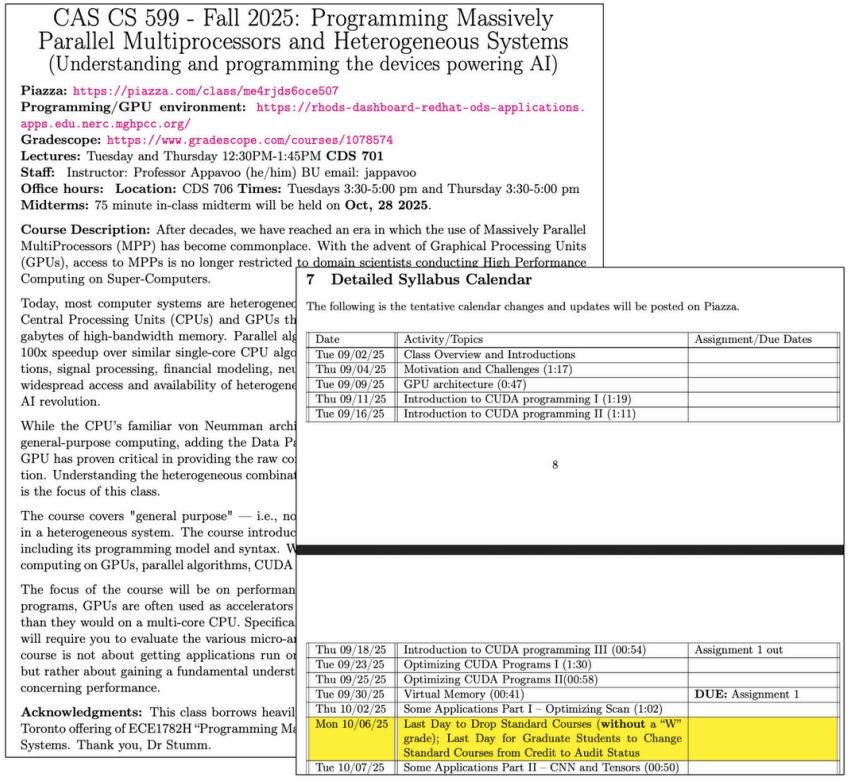
Grade break down

- Programming Assignments:
  - 2 x 15% = 30%
- Midterm:
  - **Oct 28th**: 20%
- Group Project
  - 50 %
  - 3-4 Students
  - Proposal, Midway Review, Report and Presentation

Syllabus

the details, including a tenative weekly calendar

Please <u>read</u> and ask any questions you have on Piazza



CAS CS 599 - Fall 2025: Programming Massively Parallel Multiprocessors and Heterogeneous Systems
(Understanding and programming the devices powering AI)

Piazza: https://piazza.com/class/me4rjds6oce507
Programming/GPU environment: https://rhods-dashboard-redhat-ods-applications.apps.edu.nerc.mghpcc.org/
Gradescope: https://www.gradescope.com/courses/1078574
Lectures: Tuesday and Thursday 12:30PM-1:45PM **CDS 701**
Staff: Instructor: Professor Appavoo (he/him) BU email: jappavoo
Office hours: **Location:** CDS 706 **Times:** Tuesdays 3:30-5:00 pm and Thursday 3:30-5:00 pm
Midterms: 75 minute in-class midterm will be held on **Oct, 28 2025**.

Course Description: After decades, we have reached an era in which the use of Massively Parallel MultiProcessors (MPP) has become commonplace. With the advent of Graphical Processing Units (GPUs), access to MPPs is no longer restricted to domain scientists conducting High Performance Computing on Super-Computers.

Today, most computer systems are heterogenec
Central Processing Units (CPUs) and GPUs th
gabytes of high-bandwidth memory. Parallel alg
100x speedup over similar single-core CPU algo
tions, signal processing, financial modeling, neu
widespread access and availability of heterogene
AI revolution.

While the CPU's familiar von Neumman archi
general-purpose computing, adding the Data Pa
GPU has proven critical in providing the raw co
tion. Understanding the heterogeneous combinat
is the focus of this class.

The course covers "general purpose" — i.e., no
in a heterogeneous system. The course introduc
including its programming model and syntax. W
computing on GPUs, parallel algorithms, CUDA

The focus of the course will be on performan
programs, GPUs are often used as accelerators
than they would on a multi-core CPU. Specifica
will require you to evaluate the various micro-a
course is not about getting applications run o
but rather about gaining a fundamental underst
concerning performance.

Acknowledgments: This class borrows heavil
Toronto offering of ECE1782H "Programming Ma
Systems. Thank you, Dr Stumm.

7 Detailed Syllabus Calendar

The following is the tentative calendar changes and updates will be posted on Piazza.

Date	Activity/Topics	Assignment/Due Dates
Tue 09/02/25	Class Overview and Introductions	
Thu 09/04/25	Motivation and Challenges (1:17)	
Tue 09/09/25	GPU architecture (0:47)	
Thu 09/11/25	Introduction to CUDA programming I (1:19)	
Tue 09/16/25	Introduction to CUDA programming II (1:11)	

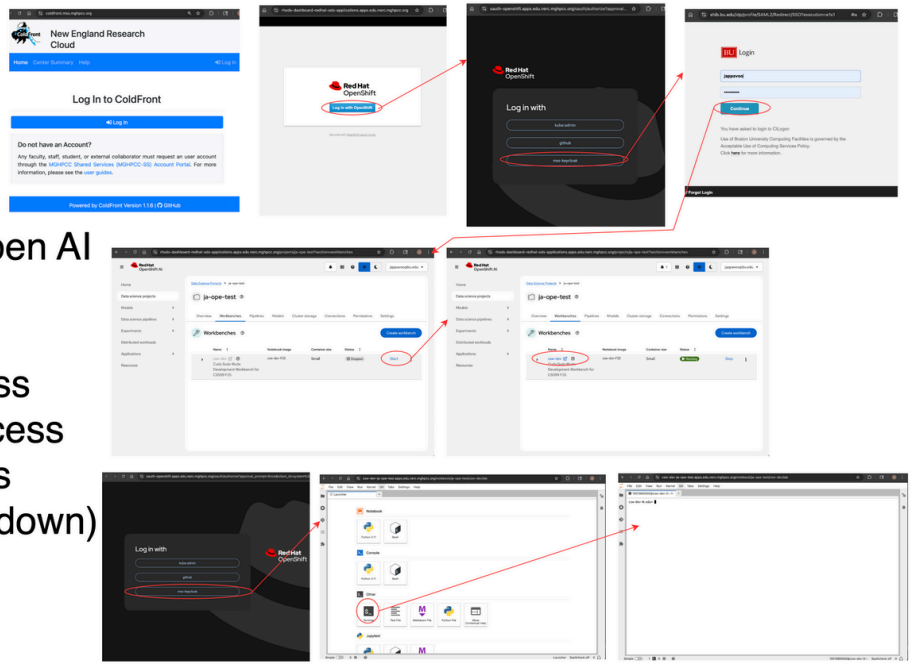
8

Thu 09/18/25	Introduction to CUDA programming III (00:54)	Assignment 1 out
Tue 09/23/25	Optimizing CUDA Programs I (1:30)	
Thu 09/25/25	Optimizing CUDA Programs II(00:58)	
Tue 09/30/25	Virtual Memory (00:41)	**DUE:** Assignment 1
Thu 10/02/25	Some Applications Part I – Optimizing Scan (1:02)	
Mon 10/06/25	Last Day to Drop Standard Courses (**without** a "W" grade); Last Day for Graduate Students to Change Standard Courses from Credit to Audit Status	
Tue 10/07/25	Some Applications Part II – CNN and Tensors (00:50)	

GPU / Lab Infrastructure

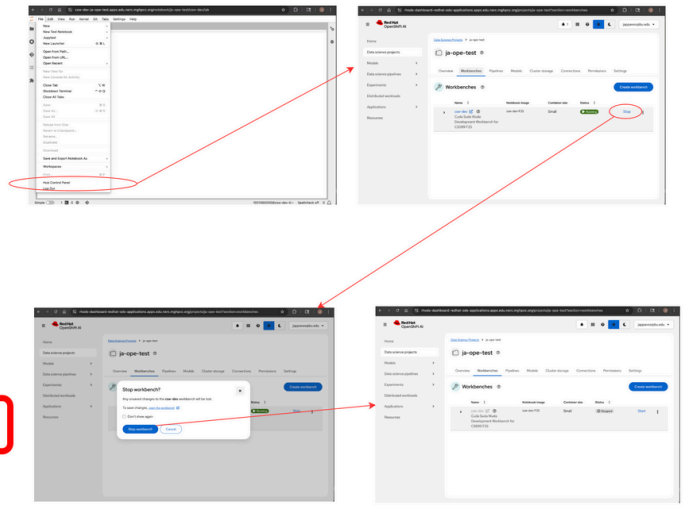
Containerized CUDA + Batch GPUs

- Getting Started
  - ColdFront
  - OpenShift and Red Hat Open AI
- Dev Container
  - RHEL 9 + CUDA 12.6.3
  - Browser Jupyter Lab Access
  - Terminal Remote Shell Access
  - Persistent home directories
  - Idle culler (but please shutdown)
- Batch Queues
  - bqstat, bjobs, brun,



Containerized CUDA + Batch GPUs

- Getting Started
  - ColdFront
  - OpenShift and Red Hat Open AI
- Dev Container
  - RHEL 9 + CUDA 12.6.3
  - Browser Jupyter Lab Access
  - Terminal Remote Shell Access
  - Persistent home directories
  - Idle culler (but please shutdown)
- Batch Queues
  - bqstat, bjobs, brun,

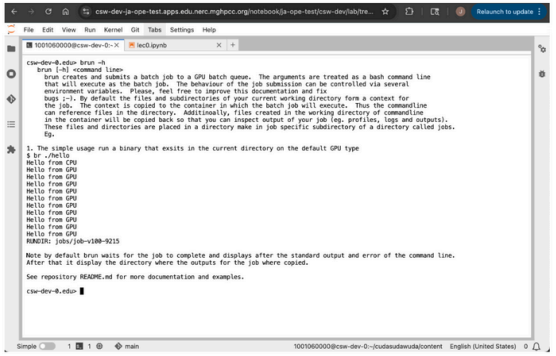


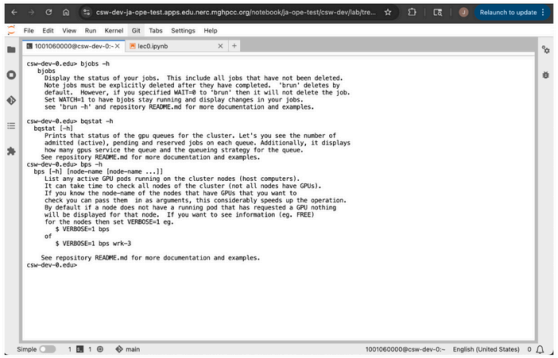
Batch Queues

https://github.com/jappavoo/batchtools

- **`brun`**: run a binary on a GPU "node"
- **`./jobs`**: outputs from run/job
- **`bjobs`**: current jobs
- **`bqstat`**: status of queues
- **`bps`**: list activity on GPU nodes (broken)
- misc: **`blog`**, **`bpods`**, **`bwait`**

Please help me fix bugs



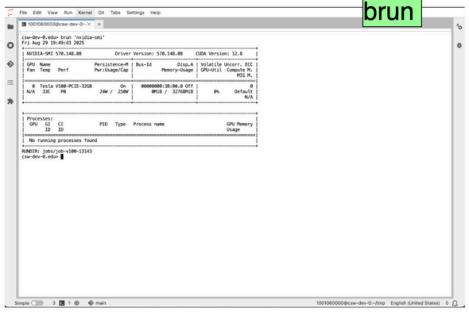


Please help me fix bugs

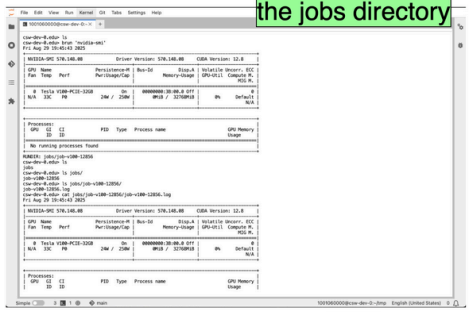
Batch Queues

https://github.com/jappavoo/batchtools

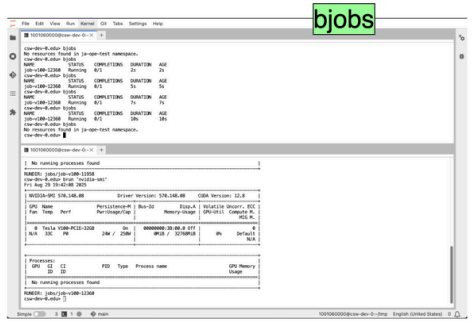
brun



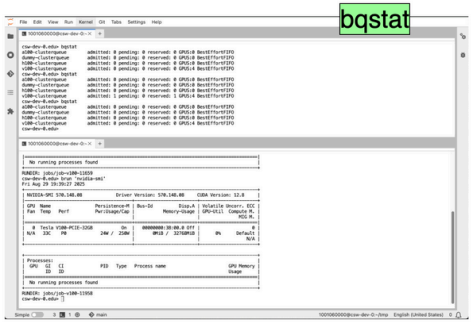
the jobs directory



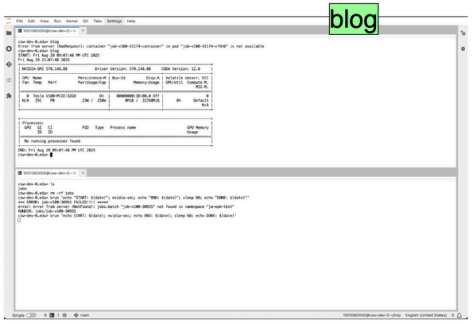
bjobs



bqstat



blog



bps

Demo

Please confirm that you can log in and run "nvidia-smi".

Terminal Interface

Alternative to using browser

- Install OpenShift CLI client (oc)
  - https://console.apps.edu.nerc.mghpcc.org/command-line-tools
- Get and execute login command
  - https://oauth-openshift.apps.edu.nerc.mghpcc.org/oauth/token/display
- Start csw-dev workbench via Red Hat OpenShift AI
- Use 'oc rsh' and friends
  - see my csw script for an example of how I start shells sessions

Concurrency, parallelism, scalability, me and you

Our backgrounds, our understanding and why we are here

Some Ice Breakers

Well for geeks anyway

- What does "Parallel Computing" make you think/feel?
- How many cores does your laptop have?
- What is the largest number of cores that a system you have worked on has had?
- What is the largest number of threads you have explicitly created in a program?
- Have you written a pthreaded application? How long did you debug it?
- What are your CS interests?
- Why brings you to this class?