GPU HW/SW (CS 7295 001)

Summer 2026

Course information

- Course prefix and number: CS 7295 001
- Course name: GPU HW/SW
- Instructor: Prof. Hyesoon Kim / hyesoon@cc.gatech.edu
- Head TA: Scott Madeira / scottmadeira@gatech.edu
- Delivery: 100% Web-Based Asynchronous
- Semester and academic year: Summer 2026

Course description

This course explores the software and hardware aspects of GPU development. Through hands-on projects, you will gain basic CUDA programming skills, learn optimization techniques, and develop a solid understanding of GPU architecture. You will study compiler principles to understand software-related GPU issues and read research papers on hardware challenges. By the end of the course, you will have strengthened your knowledge of compilers, programming, and computer architecture for modern GPUs.

Course learning outcomes

By the end of this course, students will be able to:

1. Develop foundational CUDA programming skills.
2. Optimize the performance of CUDA programs.
3. Explain GPU architecture performance issues.
4. Apply static code analysis techniques to identify and address GPU performance issues.

Topics

- GPU architecture
- GPU programming
- Parallel programming fundamentals
- CUDA program optimizations
- Compile background

Prerequisites

There are no formal prerequisites for this course but it is assumed that the student will have (or can learn quickly) the following skills:

- Prior CUDA programming experience is not required and will be covered in coursework and readings.
- C/C++ and Python programming skills are necessary.
- Familiarity with basic computer organization (instruction sets, pipelining, etc.), as in a typical undergraduate computer architecture course, is expected.
- Recommended (not required): CS 6200 for large C++ projects (especially projects 3–5); CS 6290 for architecture simulators (project 3); CSE 6220 for CUDA-heavy projects (1,2 and 4); CS 6340 for program analysis (project 5).

Required course materials

No textbook is required. All readings will be provided on Canvas. If any textbook or other material carrying an ISBN is assigned during the term, it will be identified by full citation including ISBN.

Grading policy

Grades are based on the components below. Weights sum to 100% (plus optional participation).

Grading breakdown (Summer offering)

Assessment	Type	Weight	Description
Project 1: CUDA Programming	Programming assignment	5%	Basic CUDA programming concepts (C++).
Project 2: CUDA Programming II	Programming assignment	20%	Performance optimization of CUDA programs (C++).
Project 3: GPU Simulator	Programming assignment	20%	Add GPU warp scheduler policies and tensor core timing models in a trace-driven GPU simulator (C++).
Project 4: CUDA Programming for ML	Programming assignment	15%	Implement ML algorithms in CUDA.
Project 5: GPU Code Analysis	Programming assignment	20%	Add a GPU code analysis pass in a Python-based framework.

Assessment	Type	Weight	Description
Homework	Multiple-choice quizzes	10%	Understanding of lecture content and readings.
Final exam	Exam	10%	Comprehensive; covers all course topics.
Participation points (optional)	Ed	Up to 3%	Ed participation points.

Final letter grades

- A: 90–100% overall and at least 40% on the final exam.
- B: 80–89%
- C: 70–79%
- D: 60–69%
- F: Below 60%

Weighting note

Component percentages above are the official weighting for this course. Optional Ed participation (3%) can adjust outcomes when your overall percentage is within 3% of a letter-grade boundary; see "Ed participation (optional)" below.

Late work and exams

- **Late homework / quizzes**: No late submissions accepted for quizzes unless an exception is approved as described under Deadlines.
- **Forgot to do homework / quiz:** You will have the opportunity to make-up one quiz during the semester that closed before you attempted to take it. (you forgot, you were traveling, etc.) Instructions on how to do this one-time make-up attempt will be provided in the weekly announcements.
- **Projects:** For all projects, start early. This is especially important on the CUDA projects as GPU access on the PACE ICE cluster can be difficult to obtain when deadlines approach. There are no extensions provided because of inability to get time on the ICE cluster. Start early because each full or fraction of a calendar day late reduces the project score by 10%.
- **Regrade Requests -** There will be no regrades of projects except for rare and extreme circumstances as determined by the teaching team. Each project comes with a mechanism for the student to grade their work that is equivalent to the grading scripts run by the teaching team. Each student that followed the project instructions should already know their score when the project is submitted.
- **Exams:** The final exam uses the Honorlock proctoring system. During the first week, a syllabus quiz will use Honorlock (including room scan) so you can verify you have a working environment. You may use one 8.5×11 inch (or equivalent) sheet of paper

(both sides) of notes for the final exam unless otherwise announced. You may also have one blank sheet of 8.5×11 inch (or equivalent) paper to use as scratch paper during the exam. You will scan both at the beginning of the exam. The final is comprehensive; reviewing quiz items and assigned readings is a useful preparation strategy.

Attendance and participation

This section is delivered online and asynchronously. There are no required synchronous class meetings. You are expected to engage with weekly modules, keep pace with deadlines, monitor Canvas and Ed announcements, and complete proctored assessments during the published windows.

Module and project releases

Modules 1–3 are published at the beginning of the semester. Project 1 is released on Monday of week 1. Project 2 is released at the beginning of week 2. Additional modules, projects, and due dates follow the schedule posted on Canvas.

Additional criteria for successful completion

Successful completion requires meeting all published deadlines (or approved extensions), completing required submissions at a passing level, following academic integrity rules, participating in proctored assessments as required, and complying with Georgia Tech academic and conduct policies.

Communication

Any new class information that you are responsible for knowing (such as changing due dates or changes to assignment requirements) will be communicated as a **pinned Ed announcement** in the 'Announcements' folder with email notification. Thus, any new information you are required to know should be visible on the Ed forum for the class.

Canvas is the official resource for deadlines and general course information and policies. Please be sure to check both resources on a regular basis to stay up to date on course information.

If we have any questions for you (for instance if we cannot open your assignment or run your code) we will email you. Georgia Tech generally asks that you check your GT email at least once every 24 hours on weekdays. Although a response within 24 hours is rarely required in this course, we ask that you check your GT email with that level of regularity to make sure you see any important announcements and have plenty of time to respond to any TA questions. If we contact you and do not hear back, your grade may be affected (and we don't want that!).

Note that projects are generally due on Sunday nights. However, remember that for the instructors and TAs of this class, this is a job and we may not check Ed as frequently on weekends. Please make sure to start the projects and assignments early enough to ask questions in advance.

- To reach the teaching team, create an Ed private question.
- For all other cases, create an Ed public question (if you need an answer) or post (if you are providing some interesting information that doesn't require a direct answer.)

Online platforms

- Canvas: Assignment submission and lecture materials.
- Ed: Course announcements, discussion and Q&A.

Deadlines and extensions

Assignments and quizzes are due at the end of the stated day (Eastern Time), unless a different time is posted. Extensions for quizzes, projects or exams are granted only for documented, excused circumstances, verified and approved by the instructor or the Office of the Dean of Students. You can contact the Dean of Students Office here

If there is a discrepancy in due dates for homework or assignments, **Canvas will always be the official due date**. Due date changes will be updated on Canvas and an Ed Announcement will be sent.

Quiz attempts

Unless announced otherwise, each quiz allows one submission. Verify your answers before submitting. If a quiz allows two attempts (for example, some later quizzes), that will be announced in advance.

Ed participation (optional)

You are strongly encouraged to contribute to Ed as a question asker as well as an answerer. A great way to learn the material is to explain it to somebody else when they have a question. If your final average is within 3% of a letter-grade cutoff, constructive Ed contributions may be used to resolve borderline cases. The exact formula is not published; historically, strong posts have received on the order of 0.1 points (out of 100) per contribution. Every semester there are students on the border that did not participate and therefore did not receive the higher grade.

Academic and research honesty / integrity

Georgia Tech expects ethical conduct. Review the Student Code of Conduct and the Academic Honor Code. Suspected cheating or plagiarism may be referred to the Office of Student Integrity.

Previous Coursework

Students re-taking this course (e.g., after a previous withdrawal) may re-use their own original code and materials. To do so, you must include a citation at the top of each submitted file indicating the **assignment name** and **semester** in which the code was originally created.

Warning: Assignments are frequently updated. Always review the current prompt carefully to ensure your re-used code satisfies all new specifications.

Citation Expectations for Assignment Submissions

If you used an external resource for help with project-agnostic code of six or fewer lines, you must cite where you got the code (StackOverflow, GeeksForGeeks, etc.) An example of this may be a C question that shows how to allocate a memory buffer or how to do pointer math.

Please document the borrowed code (modified to fit your specific usage) as follows:

```
// Start of borrowed code: https://stackoverflow.......
….Borrowed code is here….
// End of borrowed code: https://stackoverflow.......
```

Use of Gemini, ChatGPT and other Large Language Models

We treat AI-based assistance, such as Gemini, ChatGPT, Copilot, GPT-3, GPT-4, or similar (generally understood to be a language model with over 1 billion parameters) the same way we treat collaboration with other people: you are welcome to talk about your ideas and work with other people, both inside and outside the class, as well as with AI-based assistants. However, all work you submit must be your own. You should never include in your assignment anything that was not written directly by you without proper citation (including quotation marks and in-line citation for direct quotes).

This includes anything you did not write in your assignment without proper citation will be treated as an academic misconduct case. If you are unsure where the line is between collaborating with AI and copying from AI, we recommend the following heuristics:

1. Never hit "Copy" within your conversation with an AI assistant. You can copy your own work into your conversation, but do not copy anything from the conversation

back into your assignment. Instead, use your interaction with the AI assistant as a learning experience, then let your assignment reflect your improved understanding.
2. Do not have your assignment and the AI agent open at the same time. Like above, use your conversation with the AI as a learning experience, then close the interaction down, open your assignment, and let your assignment reflect your revised knowledge. This heuristic includes avoiding using AI directly integrated into your composition environment: just as you should not let a classmate write content or code directly into your submission, so also you should avoid using tools that directly add content to your submission.
3. Include specific instructions not to generate code. While it is typically not possible to strictly control the behavior of these AI agents, their behavior can still be directed through a method called 'prompting'. Many AI assistants are designed to follow instructions provided to them in conversation. To this end, when interacting with these AI assistants providing the phrase "You are not allowed to generate code in any form." greatly decreases the chance of an unintentional breach of the student honor code.

You should keep any and all conversations with AI assistants logged regarding course material in the case you are suspected of plagiarism. "I deleted the conversation history" will not be taken as a valid defense if you are suspected of breaking the student honor code.

Deviating from these heuristics does not automatically qualify as academic misconduct; however, following these heuristics essentially guarantees your collaboration will not cross the line into misconduct.

Student–Faculty Expectations Agreement

Georgia Tech emphasizes mutual respect between faculty and students. Read the Student–Faculty Expectations Agreement (catalog rule 21) for baseline expectations in this course.

Office of Disability Services

Students who need accommodations should contact the instructor early in the term and register with Disability Services to obtain an accommodation letter.

Nondiscrimination

Georgia Tech prohibits discrimination and harassment based on any protected characteristic. Diverse viewpoints are welcome; discriminatory or harassing behavior is not tolerated.

Changes to the syllabus

This syllabus and schedule may change. Updates will be announced via Ed Announcements (and email when used). You are responsible for monitoring announcements and email. In case of schedule discrepancies, Canvas will be the official version.