Carleton University, Computer Science, Fall 2024

Comp 1008 Math for Game Programmers

Lectures/Instructor: Thursday 6-9pm, Check on Carleton Central for the room location, **W.R. Lalonde**.

Weekly Assignments: Due Wednesday at Midnight (for assignment provided the week before).

Course Goal: Provides the math background for subsequent game programming courses.

Course calendar description: Math for building 3D games. Points, vectors, normals. Dot and cross products. Transformations and inverses in left- and right-handed systems. Uses for controlling objects, cameras, and texture manipulation. Bounding boxes, planes, frustums for collision detection and visibility, fast billboarding techniques, point and sphere sweeping. Quaternions.
Prerequisite(s): One Grade 12 university preparation mathematics course.

Course Work: 1 weekly assignment (70%) – 10 total, in class paper test (10%), and in class paper exam (20%). Available from https://brightspace.carleton.ca/ in COMP1008A

Brightspace Course materials: Powerpoint notes; **Pdf NOT USED since slides are animated**.

Book: Reference (not required) **Mathematics for 3D Game Programming & Computer Graphics (Third edition is the latest)**, Eric Lengyel, Charles River Media, Inc.

Handing in: Assignment files should be handed into Brightspace before midnight on Wednesday (for assignment provided the week before). Brightspace may not accept late assignments. Files can be PowerPoint, Word, or Text files (the advantage of PowerPoint is it allows diagrams).

Discord Link: https://discord.gg/2vwYZXtQ5y (icon contains C1-A) I'm not on the site.

E-mail: wilf.lalonde@gmail.com (I'll be looking for e-mails on Wednesday's because of assignments)

Test and Exams: Held in class Oct 8 Test (1 hour) + Dec 4 Exam (3 hours) on paper (no notes allowed).

TA (Marking only): Not yet known

SCS Laptop Requirement (applies to on-campus courses)

Every student that has been enrolled in a 1000-level (i.e., first year) course offered by the School of Computer Science after the 2020/2021 school year is required to have a laptop. For more information, please visit https://carleton.ca/scs/scs-laptop-requirement/ and then review the requirements at https://carleton.ca/scs/scs-laptop-requirement/laptop-specs/.

Course Outline

The rendering pipeline

- Generic overview
- Left versus right-handed coordinate systems
- Left-to-right versus right-to-left evaluation math
- Coordinates spaces (object, world, camera, and perspective)

Tuples

- The distinction between points, vectors, normals
- Two definitions of vector dot product
- Two definitions of vector cross product
- Intuitions behind dot products and cross products
- Many operations and related theorems on tuples, points, vectors, dot products and cross products.

Matrices

- Matrix multiply, matrix transposes, and vector-matrix operations
- Matrix inverses and how to compute them
- Useful theorems involving transposes and inverses
- Matrix forms of dot and cross products

Transformations

- Translating, rotating, scaling transformations and their inverses
- Projection transformations
- Properties of rotations
- The general rotation transformation
- Fast inverses

Transformations for gaming

- Placement matrices versus Delta matrices in the context of rotating (R), scaling (S), and translation (T).
- Controlling placement by using the natural order SRT
  - Object placement in worlds
  - Texture placement in textured objects
  - Camera placement in worlds
- Controlling and animating changes via delta matrices
  - Changes relative to an Object (pre-transformations)
  - Changes relative to a World (post-transformations)
  - Changes relative to a Parent
- Controlling changes when dealing with inverses
- Placement and Delta Changes in right to left-handed systems
- Articulated figures, poses, and skinning
- Object and camera “look at” functions

Visibility determination

- Bounding boxes, planes, frustums, and visibility trees
- Octrees, quad trees, bounding box trees, and bsp trees.
- Distance to planes, plane transformations
- Building frustums, frustum transformations
- Determining if points, bounding boxes, and spheres are inside frustums
- Sprites and how to draw them without needing to rotate them
- Portal visibility

Collision detection and collision reacting

- Movement boxes and collision detection trees (similar to visibility trees)
- Basic algorithms for projections of points on a line, on a plane
- Basic algorithm for intersection of a line with a plane, with a sphere
- The concept of object sweeping.
- Point sweeps colliding with planes, spheres, bounding boxes, polygons, and polygon soup.
- Sphere sweeps colliding with planes, spheres, bounding boxes, polygons, and polygon soup.
- More general sweeps done efficiently

Odd and ends

- Quaternions if time admits

Course Takeaway

Deep intuitive and mathematical understanding of vectors and transformations. Less mathematical but still intuitive understanding of visibility and collision detection.

Artificial Intelligence

This course was designed to be completed by individuals working alone. For this reason, tests and exams are given in class without in-class access to external tools or notes. For assignments, you are allowed to use large language models (LLMs), but you are responsible for their incorrect answers.

University Policies

Academic Accommodations

Carleton is committed to providing academic accessibility for all individuals. Please review the academic accommodation available to students here: https://students.carleton.ca/course-outline/.

Student Academic Integrity Policy. Every student should be familiar with the Carleton University Student Academic Integrity policy. A student found in violation of academic integrity standards may be sanctioned with penalties which range from a reprimand to receiving a grade of F in the course, or even being suspended or expelled from the University. Examples of punishable offences include plagiarism and unauthorized collaboration. Any such reported offences will be reviewed by the office of the Dean of Science. More information on this policy may be found on the ODS Academic Integrity page: Academic Integrity | Faculty of Science (carleton.ca).

Plagiarism. As defined by Senate, "plagiarism is presenting, whether intentional or not, the ideas, expression of ideas or work of others as one's own". Such reported offences will be reviewed by the office of the Dean of Science. More information and standard sanction guidelines can be found here: https://science.carleton.ca/students/academic-integrity/. Please note that content generated by an unauthorized A.I.-based tool *is* considered plagiarized material.

"If you are unsure of the expectations regarding academic integrity (how to use and cite references, if unauthorized collaboration with lab- or classmates is permitted (and, if so, to what degree), then you must ASK your instructor. Sharing assignment or quiz specifications or posting them online (to sites like Chegg, CourseHero, OneClass, etc.) is ALWAYS considered academic misconduct. You are NEVER permitted to post, share, or upload course materials without explicit permission from your instructor. Academic integrity offences

are reported to the office of the Dean of Science. Information, process and penalties for such offences can be found on the ODS webpage: https://science.carleton.ca/students/academic-integrity/*."*