

Data Structures and Algorithms for Processing Geometry



Course Description

VGP393A

Fall 2007, 3 credits

Saturday, 9:00AM - 12:45PM

Room #201

In this course students will learn fundamental data structures and algorithms used in the processing of 3D geometry and their applications to game development. By the end of the course, students will be familiar with key data structures and algorithms in the following areas:

- Bounding volumes
- Primitive intersection tests
- Space partitioning
- Polygon storage and management
- Numerical stability / geometric robustness

The complete, up to date, course syllabus is also available on-line at the course website (<http://people.freedesktop.org/~idr/2007-VGP393/>). The syllabus is available as both HTML and PDF (<http://people.freedesktop.org/~idr/2007-VGP393/VGP393.pdf>).

Prerequisite

This course is very programming intensive. A strong background in C or C++ programming is required. Familiarity with common data structures such as linked lists, stacks, queues, and binary trees is also required. Some background in matrix math and trigonometry will be helpful but are not required.

Texts

Required text:

Christer Ericson. *Real-Time Collision Detection*. Morgan Kaufmann, December 2004. ISBN 1558607323.

The book also has a website (<http://realtimecollisiondetection.net/>), that includes errata and some additional references.

Required Materials

In addition to paper and writing utensils, each student will need a removable storage device. The storage device will be used to both bring documents and sample code home from class and bring homework completed assignments to class. The storage requirements should be minimal, so a small USB flash-drive (256MB) should be sufficient.

Grading

Each student's grade in this course will be primarily based on a total of five single-week programming assignments and one four-week programming project. The remainder of the student's grade will be based on a mid-term exam, a final exam, and an in-class presentation.

Programming assignments will be graded first and foremost on whether or not correct output is produced. The remaining points are based on the style of the program. This includes, but is not limited to, algorithm selection, code formatting, and naming conventions. A detailed rubric will be provided with each assignment.

Programming Assignments

In-class presentation	20 pts.
Homework programming assignments	50 pts.
Term project	50 pts.
Subtotal	120 (64%)

Tests

In-class quizzes	20 pts.
Final Exam	50 pts.
Subtotal	70 pts. (36%)
Total	190 pts. (100%)

Some assignments *may* carry extra-credit opportunities, but they will be infrequent.

Grading Scale

A	=	93% and above
A-	=	90%-92%
B+	=	87%-89%
B	=	83%-86%
B-	=	80%-82%

C+	=	77%-79%
C	=	73%-76%
C-	=	70%-72%
D+	=	67%-69%
D	=	60%-66%

Late Work

I do not accept late work. If you miss a deadline, you will not earn the points for that activity. There are no make-up opportunities. If you are unable to attend class on the due date for a assignment, please submit it by e-mail *before* class.

Attendance and Participation

If you are not in class for an in-class exercise, you cannot earn those points. If you miss an entire class, you are responsible for obtaining copies of handouts and other classroom materials from your classmates.

AiPD Policies

Lab Policies

Leave food and drink outside the class. Disciplinary action will be taken toward any student found using the equipment in an inappropriate manner, taking cell phone calls or surfing the web. Disruptive, disrespectful or rude behavior will not be tolerated.

Plagiarism

Presenting the writings, images or paraphrased ideas of another as ones own, is strictly prohibited at the Art Institute of Portland. Properly documented excerpts from others works, when they are limited to an appropriate amount of the total length of a student's paper, are permissible when used to support a researched argument.

Students with Disabilities

It is AiPD policy not to discriminate against qualified students with a documented disability in its educational programs, activities or services. If you have a disability-related need for adjustments or other accommodations in this class, contact the Disability Services Coordinator.

Amber Perrin

Disabilities Services Coordinator
The Art Institute of Portland
1122 NW Davis Street
Portland, OR 97209-2911
503-382-4836
<aperrin@aii.edu>

Course Calendar

Week 1 (October 6th, 2007)

Lecture slides. (<http://people.freedesktop.org/~idr/2007-VGP393/20071006 - Bounding Volumes.pdf>)

- Course road-map
- Brief introduction to vector math
 - Addition and subtraction
 - Dot product
 - Cross product
 - Matrix multiplication
- Bounding volumes, part 1
 - What are bounding volumes, and why are they useful?
 - Axis-aligned bounding boxes (AABBs)
 - Representation
 - Intersection test
 - Creation and update
- Homework assignments:
 - Read:
Real-Time Collision Detection, Chapter 4.

Week 2 (October 13th, 2007)

Lecture slides. (<http://people.freedesktop.org/~idr/2007-VGP393/20071013 - Spheres, OBBs, and hulls.pdf>)

- Bounding volumes, part 2
 - Spheres
 - Representation
 - Intersection test
 - Creation and update
 - Oriented bounding boxes (OBBs)
 - Representation
 - Intersection test
 - Creation and update
 - Other BVs
 - Swept spheres
 - k-DOPs
 - Convex hulls
- Frustum culling and it's application for visibility testing.
- Homework assignments:
 - Bounding volumes and Frustum culling.
(http://people.freedesktop.org/~idr/2007-VGP393/20071013_Assignment.pdf) Due 10/20. (10 points)
 - Base code (<http://people.freedesktop.org/~idr/2007-VGP393/vgp393-assignment1.zip>)
 - Read:
Real-Time Collision Detection, Sections 5.1 and 5.2.

Week 3 (October 20th, 2007)

Lecture slides. (<http://people.freedesktop.org/~idr/2007-VGP393/20071020 - BV Hierarchies.pdf>)

- Bounding volume heirarchies (BVH)
 - Building a BVH
 - Bottom-up construction
 - Top-down construction
 - Incremental construction

- BVH traversal
- Sample BVHs
 - OBB trees
 - AABB trees
 - Sphere trees
- Merging BVHs
- Homework assignments:
 - BVH and Frustum culling.
(http://people.freedesktop.org/~idr/2007-VGP393/20071020_Assignment.pdf) Due 10/27. (10 points)
 - Base code (<http://people.freedesktop.org/~idr/2007-VGP393/vgp393-assignment2.zip>)
 - Read:
Real-Time Collision Detection, Chapter 6.

Week 4 (October 27th, 2007)

Lecture slides. (<http://people.freedesktop.org/~idr/2007-VGP393/20071027 - Convex Hulls.pdf>)

- Polygon representations
- Convex hulls

Week 5 (November 3rd, 2007)

Lecture slides. (<http://people.freedesktop.org/~idr/2007-VGP393/20071103 - Space partitioning.pdf>)

- Space partitioning (part 1)
 - Uniform grids
 - Octrees
 - Loose octrees
 - k-d trees
- Homework assignments:
 - Read:

Real-Time Collision Detection, Chapter 7, sections 7.1 through 7.3

- Convex hulls. (http://people.freedesktop.org/~idr/2007-VGP393/20071103_Assignment.pdf) Part 1 due 11/10, part 2 due 11/17. (20 points *total*)
- Base code (<http://people.freedesktop.org/~idr/2007-VGP393/mesh.h>)

Week 6 (November 10th, 2007)

Lecture slides. (<http://people.freedesktop.org/~idr/2007-VGP393/20071110 - Creating BSP trees.pdf>)

- BSP trees (part 1)
- Homework assignments:
 - Convex hulls, part 2. Due 11/17. (10 points)
 - Read:
Real-Time Collision Detection, Chapter 7, sections 7.4 through 7.8

Week 7 (November 17th, 2007)

Lecture slides. (<http://people.freedesktop.org/~idr/2007-VGP393/20071117 - Using BSP trees.pdf>)

- BSP trees (part 2)
- Homework assignments:
 - BSP trees. (http://people.freedesktop.org/~idr/2007-VGP393/20071117_Assignment.pdf) Due 12/1. (10 points)
 - Base code (http://people.freedesktop.org/~idr/2007-VGP393/bsp_tree.h)
 - Read:
Real-Time Collision Detection, Chapter 8.

Week 8 (November 24th, 2007)

No class this week. Thanksgiving holiday.

Week 9 (December 1st, 2007)

Lecture slides. (<http://people.freedesktop.org/~idr/2007-VGP393/20071201 - Performance.pdf>)

- Optimization
- Homework assignments:
 - TBD.

Week 10 (December 8th, 2007)

Lecture slides. (<http://people.freedesktop.org/~idr/2007-VGP393/20071208 - Numeric robustness.pdf>)

- Geometrical robustness

Week 11 (December 11th, 2007)

- Final exam. 7:45PM - 9:45PM. **Do not be late today!**