

# CSCI 397 – Spring 2007

## Game Programming

---

Ben Coleman  
coleman@cs.moravian.edu  
214 PPHAC  
Office Phone: 610-625-7781

Office Hours: MW 9:00 – 10:00  
R 9:00 – 11:00  
or by appointment  
AIM: bjcoleman15

---

### Course Description

This course focuses on the mathematics and algorithms necessary to create various types of computer games. Topics include advance programming in C++, the mathematics of game programming, artificial intelligence, event-loop programming, and 2D graphics.

### Course Goals

- Implement large programs using advanced C++ features.
- Apply patterns of design and testing to improve program development.
- Utilize trigonometry and vector mathematics to solve game-related problems.
- Apply artificial intelligence techniques to create autonomous agents.
- Use a graphics library to render appropriate visual representations of two-dimensional scenes.

### Course Texts

In addition to the following required texts, supplementary readings will be given periodically during the semester.

- *Programming Game AI by Example* by Mat Buckland
- *OpenGL, A Primer - Second Edition* by Edward Angel

You should expect to spend at least an hour before each class session working through the readings. This means reading the text for detail, not just skimming through the material before class.

### Responsibilities

Your timely attendance is expected at each class meeting. You are also responsible for the contents of reading assignments, handouts, class activities, and class email.

If you have a disability that may affect your participation in this course, please contact me immediately to discuss academic accommodations.

## Graded Materials

- Homework will be assigned nearly every class session. Some problems will be traditional pencil and paper work and others will be small programming assignments. These assignments will be due the next class session, and you should bring your solution to class. Because homework will be graded and returned the next class, you may not submit late work. You should expect to spend a half hour to a full hour on each homework assignment.
- Once per topic, I will assign programming tasks that require you to implement material from the unit. These programs will be graded based on the number of required components that function correctly. Each assignment will be given a due date, and you may not submit late work without prior permission. These programming assignments will require approximately four to six hours to complete and should not be attempted in a single sitting.
- We will implement two complete games during the semester. The first is an independent project, handed out on Friday, March 2 and due Friday, March 30. The second project will be done in a group beginning Monday, April 2 and due Friday, April 27. Further details of each game will be given later in the semester, but you should expect to spend a significant amount of time designing and implementing these games.
- Two tests will be given during the semester along with a cumulative final. The dates for the two test are Wednesday, February 21 and Wednesday, March 28.

## Academic Honesty

Except on tests, you are encouraged to discuss the material and work with other students in the course. Specifically, on homework, programming assignments, and projects, you may discuss any portion of the assignment with your fellow students. However, you must produce your own write-up of the material, and you are ultimately responsible for the material on the tests.

## Grade Determination

Homework	20%	A : 90 – 100%
Programming Assignments	20%	B : 80 – 89.9%
Projects	30%	C : 70 – 79.9%
Tests	15%	D : 60 – 69.9%
Final	15%	F : < 60%

## CSCI 397: Game Programming

### Spring 2007 Schedule

Date	Reading(s)	Topic(s)
M Jan 15		<ul style="list-style-type: none"> <li>• What does it take to write a game?</li> </ul>
W Jan 17	<ul style="list-style-type: none"> <li>• Angel pp. 1-8; 11-31; 42-44; 49-54</li> </ul>	<ul style="list-style-type: none"> <li>• OpenGL intro and basic drawing</li> <li>• Coordinate systems</li> <li>• Animation (Event loop)</li> </ul>
F Jan 19	<ul style="list-style-type: none"> <li>• Angel pp. 54-60</li> </ul>	<ul style="list-style-type: none"> <li>• Keyboard input</li> <li>• Mouse Interface</li> </ul>
M Jan 22	<ul style="list-style-type: none"> <li>• Angel pp. 139-142, 169-180</li> </ul>	<ul style="list-style-type: none"> <li>• Alpha channel</li> <li>• Texture maps</li> <li>• Using JPEG graphics</li> <li>• <b>Programming Task #1 Out</b></li> </ul>
W Jan 24	<ul style="list-style-type: none"> <li>• Buckland pp. 1-22</li> </ul>	<ul style="list-style-type: none"> <li>• Algebra</li> <li>• Trigonometry</li> <li>• Basic vector mathematics</li> </ul>
F Jan 26	<ul style="list-style-type: none"> <li>• Buckland pp. 23-25</li> </ul>	<ul style="list-style-type: none"> <li>• Dot product</li> <li>• Projection</li> </ul>
M Jan 29	<ul style="list-style-type: none"> <li>• Chapter 1 from "C++ For Game Programmers" pp. 4-24</li> </ul>	<ul style="list-style-type: none"> <li>• Inheritance</li> <li>• Polymorphism</li> <li>• <b>Programming Task #1 Due, #2 Out</b></li> </ul>
W Jan 31	<ul style="list-style-type: none"> <li>• "C++ For Game Programmers" pp. 44-63;</li> <li>• "Item 21, Use const whenever possible", pp. 91-97;</li> <li>• "C++ For Game Programmers" pp. 364-371;</li> <li>• Buckland Appendix B</li> </ul>	<ul style="list-style-type: none"> <li>• const and reference</li> <li>• Include file directives</li> <li>• UML</li> </ul>
F Feb 2	<ul style="list-style-type: none"> <li>• "The Evolution of Game AI"</li> <li>• "12 Tips from the Trenches"</li> <li>• "The Illusion of Intelligence"</li> </ul>	<ul style="list-style-type: none"> <li>• AI Overview</li> </ul>
M Feb 5	<ul style="list-style-type: none"> <li>• Buckland pp. 43-67</li> </ul>	<ul style="list-style-type: none"> <li>• State machines</li> </ul>
W Feb 7	<ul style="list-style-type: none"> <li>• Buckland pp. 67-83</li> </ul>	<ul style="list-style-type: none"> <li>• Message Passing System</li> <li>• <b>Programming Task #2 Due, #3 Out</b></li> </ul>
F Feb 9	<ul style="list-style-type: none"> <li>• gettimeofday reference - TBA</li> <li>• "The Clock: Keeping Your Finger on the Pulse of the Game"</li> </ul>	<ul style="list-style-type: none"> <li>• Frame Rate Computations</li> </ul>
M Feb 12	<ul style="list-style-type: none"> <li>• "Crashing into the New Year"</li> </ul>	<ul style="list-style-type: none"> <li>• Interior Point Computations</li> <li>• Simple Collision Detection</li> </ul>

Date	Reading(s)	Topic(s)
W Feb 14	<ul style="list-style-type: none"> <li>• “Pool Hall Lessons” Fast, Accurate Collision Detection between Circles and Spheres”</li> </ul>	<ul style="list-style-type: none"> <li>• Robust Collision Detection</li> </ul>
F Feb 16	<ul style="list-style-type: none"> <li>• Buckland pp. 28-40</li> <li>• “Collision Response” pp. 607-616</li> </ul>	<ul style="list-style-type: none"> <li>• Collision Response</li> <li>• <b>Programming Task #3 Due, #4 Out</b></li> </ul>
M Feb 19	<ul style="list-style-type: none"> <li>• “The Magic of Data-Drive Design”</li> <li>• “The Science of Debugging”</li> </ul>	<ul style="list-style-type: none"> <li>• Data-Driven Design</li> <li>• Debugging Techniques</li> </ul>
W Feb 21		<ul style="list-style-type: none"> <li>• Test #1</li> </ul>
F Feb 23	<ul style="list-style-type: none"> <li>• Buckland pp. 85-91</li> </ul>	<ul style="list-style-type: none"> <li>• Agent Behaviors - Program Layout</li> </ul>
M Feb 26	<ul style="list-style-type: none"> <li>• Buckland pp. 91-112</li> </ul>	<ul style="list-style-type: none"> <li>• “Simple” Behaviors</li> </ul>
W Feb 28	<ul style="list-style-type: none"> <li>• Buckland pp. 113-124</li> </ul>	<ul style="list-style-type: none"> <li>• Group Behaviors</li> <li>• <b>Programming Task #4 Due, #5 Out</b></li> </ul>
F Mar 2	<ul style="list-style-type: none"> <li>• Buckland pp. 125-132</li> </ul>	<ul style="list-style-type: none"> <li>• Autonomous Agent Efficiency</li> <li>• <b>Project #1 Out</b></li> </ul>
M Mar 5 – F Mar 9		<ul style="list-style-type: none"> <li>• Spring Break</li> </ul>
M Mar 12	<ul style="list-style-type: none"> <li>• Buckland pp. 133-191</li> </ul>	<ul style="list-style-type: none"> <li>• Sports Simulation Example</li> <li>• Project #1 Overview</li> </ul>
W Mar 14	<ul style="list-style-type: none"> <li>• <a href="http://cppreference.com">cppreference.com</a></li> <li>• TBA</li> </ul>	<ul style="list-style-type: none"> <li>• STL containers</li> <li>• STL iterators</li> <li>• STL algorithms</li> </ul>
F Mar 16	<ul style="list-style-type: none"> <li>• Buckland pp. 193-203</li> </ul>	<ul style="list-style-type: none"> <li>• Introduction to Graphs</li> <li>• <b>Programming Task #5 Due</b></li> </ul>
M Mar 19	<ul style="list-style-type: none"> <li>• Buckland pp. 203-209</li> </ul>	<ul style="list-style-type: none"> <li>• Graph Implementation</li> </ul>
W Mar 21	<ul style="list-style-type: none"> <li>• Buckland pp. 209-231</li> </ul>	<ul style="list-style-type: none"> <li>• Depth-First and Breadth-First Search</li> </ul>
F Mar 23	<ul style="list-style-type: none"> <li>• Buckland pp. 231-241</li> </ul>	<ul style="list-style-type: none"> <li>• Dijkstra’s Algorithm</li> </ul>
M Mar 26	<ul style="list-style-type: none"> <li>• Buckland pp. 241-247</li> </ul>	<ul style="list-style-type: none"> <li>• The A* Algorithm</li> </ul>
W Mar 28		<ul style="list-style-type: none"> <li>• Test #2</li> </ul>
F Mar 30	<ul style="list-style-type: none"> <li>• Buckland pp. 295-331</li> </ul>	<ul style="list-style-type: none"> <li>• Raven Overview</li> <li>• <b>Project #1 Due</b></li> </ul>
M Apr 2	<ul style="list-style-type: none"> <li>• Buckland pp. 333-342</li> </ul>	<ul style="list-style-type: none"> <li>• Practical Path Planning Overview</li> <li>• <b>Project #2 Out</b></li> </ul>
W Apr 4	<ul style="list-style-type: none"> <li>• Buckland pp. 342-374</li> </ul>	<ul style="list-style-type: none"> <li>• Path Planning</li> </ul>
F Apr 6 – M Apr 9		<ul style="list-style-type: none"> <li>• Easter Break</li> </ul>
W Apr 11	<ul style="list-style-type: none"> <li>• Buckland pp. 374-377</li> </ul>	<ul style="list-style-type: none"> <li>• More Path Planning</li> </ul>
F Apr 13	<ul style="list-style-type: none"> <li>• Buckland pp. 379-397</li> </ul>	<ul style="list-style-type: none"> <li>• Goal-Driven Behavior Overview</li> </ul>

<b>Date</b>	<b>Reading(s)</b>	<b>Topic(s)</b>
M Apr 16	• Buckland pp. 398-404	• Goal Arbitration
W Apr 18	• Buckland pp. 405-414	• Personalities
F Apr 20		• No class, Project Work Day
M Apr 23	• TBA	• TBA
W Apr 15	• TBA	• TBA
F Apr 27		• Review • <b>Project #2 Due</b>