

# Game Physics

Fall 2015

Gam 350

**instructor:** Ed Keenan  
**email:** [ekeen2@cdm.depaul.edu](mailto:ekeen2@cdm.depaul.edu)  
**office hours:** Wed 1-4pm 14EAS 505, 30 min before class or by email appointment  
**office:** CDM 830  
**phone:** (312)362-6747  
**website:** [piazza.com/depaul/fall2015/gam450](http://piazza.com/depaul/fall2015/gam450) (Preferred communication)  
**lecture:** LEWIS 1005, Mondays, 5:45-9:00pm  
**Desired to Learn (D2L):** [d2l.depaul.edu](http://d2l.depaul.edu) (Grades, Viewing lectures, Announcements)  
**Version Control:** perforce: **140.192.39.61:1666**

## Description:

The course concentrates on Newton's Laws of Motion, kinematics and kinetics. This theory will be applied to problems that a game programmer must understand e.g. collisions between objects, projectiles and their trajectories, real-time simulation of motion. Special objects such as cars, aircraft and ships will be discussed. Students will apply and implement laws of physics.

## Prerequisites:

- Data Structures in Java or C++ (CSC 301 or CSC 383 or CSC 393)
- C++ programming language (CSC 309 or CSC 362)
- Vector Matrix Math (GAM 321)
- or instructor consent

## Learning Goals:

- Students will be able to integrate and use a commercial 2D physics engine.
- Students will be able to identify and modify the components of real-time game physics simulations, such as bodies, integration steps, contact manifolds, fixtures, collisions, impulses and forces.
- Students will be able to understand the relationship between continuous time based and discrete time physics systems.
- Students will be able to analyze game objects and modify the corresponding collision fixtures and manifolds during physics simulations to produce the desired interactive effect.
- Students will be able to create programmatically bounding volumes for 3D objects.
- Students will be able to apply 3D vector and matrix math

## Grading:

30 % Box2D Demos ( best 6 out of 8 scored )

- PA1 - (5%) Sprite engine
- PA2 - (5%) Concave Bounding Volumes
- PA3 - (5%) Multiple Collisions
- PA4 - (5%) Ghost Vertices (hardest)
- PA5 - (5%) Particles
- PA6 - (5%) Rotation Joints
- PA7 - (5%) Explosions (hardest)
- PA8 - (5%) Rope Joints

60% Projects

- Milestone 1: (25%) Angry Birds - Basics
  - (20%) Required Features
  - (5%) check pts
- Milestone 2: (35%) Angry Birds - Advanced
  - (30%) Required Advanced Features
  - (5%) check pts

10% Piazza / Perforce Changelist {priceless}

## Textbooks and printed resources

### Required Texts:

- **None - required**

### Optional:

- **Box2D for Flash Games**
  - Feronato, Packt Publishing 2012. ISBN: 978-1849519625
- **Real-Time Collision Detection**
  - Ericson, Morgan Kaufmann/Elsevier Science, 2005. ISBN: 978-1558607323

### Additional material provide in class:

- Will be provided by the instructor
- Lectures, links, SDKs and other corresponding material

## Software

- **Microsoft Visual Studio 2013 (ultimate edition - Recommended)**
  - [DePaul MSDNAA link Microsoft Visual Studio Ultimate 2013](#)
  - C++ and C# install (future classes)
  - Microsoft Visual Studio 2015 is not used in this class.
- Perforce - Visual Client (p4v)
  - [www.perforce.com](http://www.perforce.com)
- Download and configuration instructions will be provided in class
  - Server address: **140.192.39.61:1666**
- Box2D - 2D Physics Engine for Games
  - [www.box2d.org](http://www.box2d.org)

## Box2D demos (prototypes):

All Box2D demos are mini prototypes stand alone projects to shake out a concept or a problem. The goal of these mini-problems, quickly get to the heart of the problem: create a demo and solve the core issues. Once this is done, these solutions can be easily adapted into other projects (such as Angry Birds).

Requirements of all Demos:

- Stand-alone project (source code, project and resources)
- YouTube Video demo with commentary of the video

### PA1 - Sprite engine / Box2D integration

- Understanding of the real-time C++ sprite engine
- Create multiple sprites using Angry Birds textures
- Moving and bouncing around on the screen

### PA2 - Concave Bounding Volumes

- Creating concave collision objects
- Demonstrating the concave object reacting in an environment (colliding and reacting)
- Constructing object as a collection of several convex object welded together

### PA3 - Multiple Collisions

- 1 moving object *simultaneously* hitting 2 static collision objects
- Registering 2 separate collisions

### PA4 - Ghost Vertices

- Collision object following or dragging on top of terrain objects (boxes)
- Object getting stuck or colliding resulting in discontinuous behavior

### PA5 - Particles

- Spawning several many small objects from a point source
- Point source may be static or moving
- Particles with unique attributes and lifetimes

### PA6 - Rotation Joints

- Planetary Gears
- Objects attached to other objects at specific point
- Rotation object having constraints - angle, velocity, etc...
- Series of rotation objects to create snake and chain segments

### PA7 - Explosions

- Applying proportional impulses base on
- Raycast method
- Proximity method
- Particle method

### PA8 - Rope Joints

- 2 objects attached together by flexible rope
- Allowing swinging and movement restriction (length of string)
- Rigid string or Rods allowing fix arc motion

# Angry Bird Project

## Milestone 1: Basics Features

Simple basic Angry birds levels using only the default red birds with simple blocks. Complete details to be distributed in class as part of Milestone 1.

**Quick summary:** (all students)

- Create 2 demos - Poach Eggs copy
  - Theme: Level 1-1
  - Theme: Level 1-9
- Red Bird
  - Sling shot, moving with kinetic motion
- Blocks
  - Glass, Wood and Rock with several levels of sprites representing damage / strength
- Pigs
  - Collidable object that disappears when damage, show different damage through sprites.
  - With/without helmets
  - Animated
- Sound Effects
  - Collision events
- Graphics
  - Used supplied sprite sheet for sprite art
  - Background art

Progress Drops

- Milestone 1 check points
- Quick check point on progression
  - Must include 30-60 sec YouTube clip with commentary
  - Checklist document
  - Source code submission

Final Drop

- Code drop
- 3 - 5 min YouTube clip demo with commentary

## Milestone 2: Advanced Features

Advanced Angry birds levels using the Red, Blue, White, Yellow, Green and Black birds to mimic the commercial game. More advanced blocks, circular, rectangular and triangular shapes. Animation on Pigs with simple blocks. Complete details to be distributed in class as part of Milestone 2.

### Quick summary:

#### Required Features (all students)

- Create 2-4 demos - Poach Eggs copy
  - Show case the features
  - Can be several independent demos
- Red, Blue, White, Yellow, Green Birds
  - Specialize flight and spawning behaviors
- Blocks
  - Glass, Wood and Rock with several levels of sprites representing damage / strength.
  - Different shapes - circles, triangles, rectangles
- Pigs
  - Collidable object that disappears when damage, show different damage through sprites
- Sound Effects
  - Collision events
  - Ambient
  - Music background
- Graphics
  - Used supplied sprite sheet for sprite art
  - Background art
  - Smoke rings during flight
- Collisions
  - Collisions need to bleed energy correctly as the bird or projectile punch through several objects
- Score

#### Advanced features

- Particles
  - In flight
  - Feathers
  - Smoke
- Explosions
  - Black Bird
- Rope or Rotational objects
  - Creative way to use ropes suspended joints

#### Progress Drops

- Milestone 1 check points
- Quick check point on progression
  - Must include 30-60 sec YouTube clip with commentary
  - Checklist document
  - Source code submission

#### Final Drop

- Code drop
- 3 - 5 min YouTube clip demo with commentary

## Perforce Submissions

- Everyone is expected to submit at least 10 submissions a week to perforce.
- The biggest reason students get into trouble with software design:
  - Not working on the material frequently enough
  - Taking too large of a bite of the design
- Both are fixed with this Perforce RULE
- Even my simplest programs take 10-20 submissions.
  - For these project assignments my average is 40-400 submissions, so 10 will be no problem.
- Detailed perforce changelist comments are expected

## Piazza Discussion forum

- Statistics show: students who participate more and help other students do better!
  - The correlation is ridiculous!
    - Poor understanding / poor participation.
    - Great understanding / Great participation
  - As you master the material, help others learn!
    - You're in the master's program so master it!
- Everyone is expected and encouraged to participate on the Piazza discussion forum. All class-related discussion here this term.
- The quicker you begin asking questions on Piazza (rather than via emails), the quicker you'll benefit from the collective knowledge of your classmates and instructors. I encourage you to ask questions when you're struggling to understand a concept.
- All correspondence that is not personal in nature should be vectored through Piazza
- Sensitive material, use Piazza private note, not email.
- Keep the forum professional and positive, help each other out.
  - Karma really pays off here.
  - Help each other whenever you can.
    - There will be a section where you'll need help (trust me).

NOTE: Do NOT post until you have watched the entire lecture FIRST (in class or online)  
This will prevent frustration on all sides (members asking or answering questions)

## Collaborating together on programming assignments

- You are encourage to work together
  - Use the Piazza forums heavy
  - Even share your material with others in the common directory
- Everyone is 100% responsible for the work they do.
  - If you get help with a section of code,
  - Please refactor the code the snout out of it
    - Comment and understand that material
    - Transform the code to make it yours
  - Be able to answer any question regarding the code you commit

- System for Detecting Software Plagiarism
  - We will be using MOSS - Measure of Software Similarity (Stanford University)
    - Indicates possible code infringements (plagiarism)
    - MOSS - will detect the similarity independent of naming convention, indentation style or formatting, it compares abstract syntax tree of your code.
- If you gain significant support / help from another student
  - Fully disclose the support / help you had in a Readme.txt file submitted with your assignments.
    - Disclosing the help, is **not permission** for copying the code.
    - Only there to clarify and acknowledge help you were given from a fellow student.
- Modifying any Unit Test to alter the outcome results is also an **Academic Integrity Violation**
- If you are stuck and find yourself even tempted to plagiarize
  - Ask for help !!!!
    - Use on Piazza -> Visit during offices hours, make an appointment
  - **Don't ever compromise your integrity!**

## Tentative Class Schedule

<b>Date</b>	<b>Lecture</b>	<b>Activity</b>	<b>Due</b>
Week 1 14-Sep	Overview, Sprite Engine Box2D integration	PA1 - Sprite engine <i>* Bodies, Fixtures, World Settings</i>	
Week 2 21-Sep	Collision Shapes Contact Listeners	PA2 - Concave Volumes  <i>Milestone 1 - assigned</i> <i>* User Data, Callbacks, Filtering</i>	PA1
Week 3 28-Sep	Launching Bird Sound Engine	PA3 - Multiple Collisions  <i>* Forces/Impulses, Removing Bodies</i> <i>* Jumping</i>	PA2  <i>MS1 - check pt</i>
Week 4 5-Oct	Damage Swapping Images Sling Shot	PA4 - Ghost Vertices  <i>* Projectile Trajectory, Sticky</i> <i>* Ghost Verts</i>	PA3  <i>MS1 - check pt</i>
Week 5 12-Oct	Energy Bleeding MS1 - issues	Milestone 1 - Angry Birds	PA4  <i>MS1 - check pt</i>
Week 6 19-Oct	Yellow Bird White Bird Particles	PA5 - Particles  <i>Milestone 2 - assigned</i> <i>* Particle Driven Physics</i>	Milestone 1
Week 7 26-Oct	Blue Bird Green Bird Smoke Trails	PA6 - Rotation Joints  <i>* Joints, Joint Revolute</i>	PA5  <i>MS2 - check pt</i>
Week 8 2-Nov	Black Bird Explosions	PA7 - Explosions  <i>* Explosions</i>	PA6  <i>MS2 - check pt</i>
Week 9 9-Nov	Hard Constraints Springs Ropes	PA8 - Rope Joints  <i>* Rope Joints</i>	PA7  <i>MS2 - check pt</i>
Week 10 16-Nov	Angry Problems 2D vs 3D Engines	Milestone 2 - Angry Birds	PA8  <i>MS2 - check pt</i>
Week 11 23-Nov			Milestone 2

September 15, 2015 Last day to add classes to AQ2015 schedule  
 September 22, 2015 Last day to drop classes with no penalty, Last day to select pass/fail option  
 September 23, 2015 Grades of "W" assigned for AQ2015 classes dropped on or after this day  
 September 29, 2015 Last day to select auditor status  
 October 27, 2015 Last day to withdraw from AQ2015 classes

## Course Policies

### Changes to Syllabus

This syllabus is subject to change as necessary during the quarter. If a change occurs, it will be thoroughly addressed during class, posted under Announcements in D2L and sent via email.

### Online Course Evaluations

Evaluations are a way for students to provide valuable feedback regarding their instructor and the course. Detailed feedback will enable the instructor to continuously tailor teaching methods and course content to meet the learning goals of the course and the academic needs of the students. They are a requirement of the course and are key to continue to provide you with the highest quality of teaching. The evaluations are anonymous; the instructor and administration do not track who entered what responses. A program is used to check if the student completed the evaluations, but the evaluation is completely separate from the student's identity. Since 100% participation is our goal, students are sent periodic reminders over three weeks. Students do not receive reminders once they complete the evaluation. Students complete the evaluation online in [CampusConnect](#).

### Academic Integrity and Plagiarism

This course will be subject to the university's academic integrity policy. More information can be found at <http://academicintegrity.depaul.edu/>. If you have any questions be sure to consult with your professor.

### Academic Policies

All students are required to manage their class schedules each term in accordance with the deadlines for enrolling and withdrawing as indicated in the [University Academic Calendar](#). Information on enrollment, withdrawal, grading and incompletes can be found at: [cdm.depaul.edu/enrollment](http://cdm.depaul.edu/enrollment).

### Students with Disabilities

Students who feel they may need an accommodation based on the impact of a disability should contact the instructor privately to discuss their specific needs. All discussions will remain confidential.

To ensure that you receive the most appropriate accommodation based on your needs, contact the instructor as early as possible in the quarter (preferably within the first week of class), and make sure that you have contacted the Center for Students with Disabilities (CSD) at: [csd@depaul.edu](mailto:csd@depaul.edu).

Lewis Center 1420, 25 East Jackson Blvd.

Phone number: (312)362-8002

Fax: (312)362-6544

TTY: (773)325.7296

### Retroactive withdrawal

This policy exists to assist students for whom extenuating circumstances prevented them from meeting the withdrawal deadline. During their college career students may be allowed one medical/personal administrative withdrawal and one college office administrative withdrawal, each for one or more courses in a single term.

Repeated requests will not be considered. Submitting an appeal for retroactive withdrawal does not guarantee approval. Information on enrollment, withdrawal, grading and incompletes can be found at:

<http://www.cdm.depaul.edu/Enrollment-Policies.aspx>