Final Project

Published | Edit | ⋮

GPR-350 Game Physics
Instructor: Daniel S. Buckstein
Final Project

Summary:
The primary takeaways of this course are, in no particular order or rank, ***portfolio*** and ***engineering***. The final project is for you to demonstrate all you have learned in this course from a game programmer's lens and produce a portfolio-worthy project.

Submission:
Submit a link to your online repository with the completed assignment's branch name and commit ID/index. If you have not created an online repository to keep track of your work, you should do so as part of this assignment; it will be checked. ***Work in pairs***.

Instructions:

Design your own project that includes the following:

- 3D linear dynamics: integration methods for position and force generation
  - Required: explicit Euler integration for position and velocity; kinematic integration for position; at least 4 force generators (excludes 'sliding') being used at any point; Newton-2 for linear motion.
- 3D angular dynamics: integration methods for rotation, moment of inertia and torque generation
  - Required: explicit Euler integration for rotation (quaternion) and angular velocity (vector); moment of inertia for spheres and boxes; conversion of force to torque; proper implementation of Newton-2 for angular motion.
- 3D collision detection: implement data structures and algorithms required to test collisions between a variety of 3D convex hulls
  - Required: spheres, axis-aligned-boxes, non-axis-aligned boxes and at least one complex hull (e.g. made of multiple simple hulls); collision detection algorithms for

each of the above against all others.
    - Recommended: cylinders, rays (you can do so much more with these).
- 3D collision response: implement data structures and algorithms required to accurately respond to collisions between a variety of 3D convex hulls
    - Required: contacts and normal generation; physical response between spheres; at least one trigger response (non-physical; i.e. a collision event occurs and something happens).
    - Recommended: physical response between other shapes; some sort of management system to help deal with collision events.

Please run your project ideas by the instructor well in advance. Examples of types of projects to consider:

- Create a physics demo or game that is fun, winnable/losable and demonstrates all of the above.
- Demonstrate a significant extension of any of the above subject matter.
- Design a tool that helps build and/or manage the physics world and make other developers' lives easier.
- Explore a crossover of physics programming with some other game programming discipline.

All components of the project must be functional and impressive; this is something you would want to show off at an interview; hence, this project is tied with a technical interview! Find a way to demonstrate the requirements as part of the technical interview (which involves a presentation). Implementing the bare minimum will earn you up to 80% on the project; go above and beyond for the remaining 20%.

Points 20

Submitting a text entry box

Due	For	Available from	Until
-	Everyone	-	-

+ Rubric