Lab 3: Torque & Intro to Rigid Body Dynamics

Published | Edit | ⋮

GPR-350 Game Physics
Instructor: Daniel S. Buckstein
Lab 3: Torque & Intro to Rigid Body Dynamics

Summary:
This week we expand upon our particle physics by implementing torque and angular force responses. By doing this we move towards rigid body dynamics.

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:

Step 1: Moment of Inertia
Implement a private member for inertia or moment of inertia (I). This is calculated based on the shape of the particle (e.g. circle, square, box), so a drop-down would be helpful for the user to choose the type of shape a particle represents. The book has a list of 2D moment of inertia values; implement these for a variety of particles and set a particle's inertia appropriately on start (after setting mass as it is needed in all of the formulas). Much like we had for mass, we will need an inverse moment of inertia.

Step 2: Torque
Add a public torque member (float) to the particle interface. This will represent the accumulated sum of all environmental torques generated from forces applied. Torque is converted to angular acceleration using Newton-2: $\tau = I\alpha \;\longrightarrow\; \alpha = I^{-1}\tau$
Write an 'update angular acceleration' function which converts torque to angular acceleration before resetting torque. This should be called in your update after rotation integration.
Write an 'apply torque' function which adds an amount of torque to the total torque acting on the particle (D'Alembert's principle). The amount of torque applied is calcuated using the 2D

equivalent of T = pf x F: T is torque, pf is moment arm (point of applied force relative to center of mass), F is applied force at pf. It is important to note that the center of mass may not be the center of the object, so it might help to add a separate member for center of mass in local and world space.

Step 3: Test
Implement test scenarios to convert force to torque on a variety of shapes that ***do not move*** (i.e. you are only testing rotation).

Bonus:
Implement a ball that rolls (both moves and rotates) on a flat plane with user input generating a force to push it around, or on a slope that uses gravity, normal force and friction.

Points	8
Submitting	a text entry box

Due	For	Available from	Until
-	Everyone	-	-

+ Rubric