

# CS425 Game Programming 1

## Homework Assignment 6

### (Shootin' Fish into a Barrel)

Due Friday, November 11<sup>th</sup> 2016 at midnight

#### Submission:

- This is an individual programming assignment.
- When you have completed the assignment, delete the Debug directory and the .sdf file. Then zip together the rest of the directory. Be sure to include your .vcxproj and .sln files along with all of your source code files. If the project cannot be loaded and run properly, you will get a zero. Name the zipped file in the following way:  
**LastName\_FirstName\_HW6.zip** and submit it through Blackboard by the due date.

Assignment Goal: Explore simple physics simulation and collision detection.

As with the previous assignment, you can use the Unreal engine for this assignment, provided you are programming in C++ and not extensively using blueprints. This assignment write-up describes the key points in OGRE. If you choose to do the assignment in Unreal, you will have to determine the necessary Unreal method calls.

1. Implement a game where the objective is to shoot/propel fish into a barrel.
  - a. The drum model attached to this assignment can be used as the barrel and the fish model that comes with OGRE can be used as the fish.
  - b. The player should be able to modify two settings both corresponding to initial velocity of the fish. The slope of the trajectory (i.e. direction) and the speed. You can make sure that the fish and barrel are aligned so that aiming left or right isn't necessary.
    - i. Choose appropriate keyboard or mouse inputs and make sure they are documented.
    - ii. Choose appropriate camera settings.
    - iii. Include a visualization of the current settings in the game environment. For the trajectory, this might be the orientation of the fish. For speed it might be a GUI element or the scale of a cube or other primitive in the environment.
  - c. Once the initial velocity is set, the player should press the spacebar to launch the fish toward the barrel.
    - i. Physics simulation should then be used to animate the fish moving through the environment.
      1. Be sure to include the initial velocity and gravity (9.81). You can experiment with different masses for the fish.
      2. You can treat the fish as a point mass (i.e. no rotation is required).
  - d. Detect whether or not the fish intersects with the barrel before it hits the floor plane.
    - i. You are not required to do collision response.

- ii. You can use OGRE's Axis Aligned Bounding Boxes. OGRE entities have bounding boxes (See the code example from class for an example). OGRE also includes a number of *intersects* methods in the *AxisAlignedBox* class and methods for getting the corners of the bounding boxes:  
[http://www.ogre3d.org/docs/api/html/classOgre\\_1\\_1AxisAlignedBox.html](http://www.ogre3d.org/docs/api/html/classOgre_1_1AxisAlignedBox.html)
- iii. At the very least, you should determine whether the bounding box of the fish intersects with the bounding box of the barrel. It would be more accurate to do collision detection between the bounding box of the fish and the plane of the top of the barrel or perhaps a sphere representing the top of the barrel. Another option would be to hide (invisible) a cube of the appropriate size in the top of the barrel.
- iv. There should be some indication that the fish hit the barrel or did not.
  1. Perhaps turn on a particle simulation if the target was hit.
  2. Perhaps play a sound or lower the lights if the target wasn't hit.
- v. Also keep score. This could be a GUI element or adding elements to the world (e.g. particle emitters).
- e. The player should be able to repeatedly shoot fish at the barrel.
  - i. In other words, reload after each shot.
  - ii. Allow at least 20 shots per game.

Advanced features you might consider:

- Include aiming left and right.
- Have different masses for the fish (and scale them) so that each round is a new challenge.
- Include multiple barrels at different positions in the world worth a variety of points according to difficulty.
- Vary the scale of the barrel and/or fish and have it impact the score.
- Have the barrels move (i.e. moving targets) and/or rotate so that launch timing is also an element of the game.
- Include air resistance (e.g. fog or rain) and wind
- Do collision response with the floor plane and barrels (What is the COR of a fish?)
- Attach the camera to the fish as it's flying
- Treat the fish as a rigid body (i.e. include orientation)