

## Bachelor Lab Scientific Computing (Game Physics) Worksheet 4: Collisions (sphere-box, plane-box), create your own scene

### Assignment 1: Implement a sphere-box collision detection

File: *cPhysicsIntersections.cpp*,

Method: *CPhysicsIntersections::sphereBox*

Similar to a previous assignment, you should transform the center of the sphere from world space to the object space of the box. Then, all collision tests can be implemented by testing collisions with axis aligned surfaces. Fill the collision data with appropriate values!

### Assignment 2: Implement a plane-box collision detection

File: *cPhysicsIntersections.cpp*,

Method: *CPhysicsIntersections::planeBox*

There are different ways to handle these intersections. The easiest one (even if this is not the most efficient one - see next worksheet) is to transform the box into plane space:

$$M_{to\_planespace} = M_{plane}^{-1} \cdot M_{box}$$

Start by checking all vertices interpenetrating the surface of the plane.

Then test all box edges for an interpenetration with the edges of the plane. Further, the box should be able to collide with the edges of the plane. Introduce respective tests for this scenario!

### Assignment 3: Create your own scene

File: *cScenes.cpp*,

Method: *CScenes::setupScene21*

It's time to show us what you can 'design'! This assignment is quite easy: Setup a new *fancy looking* scene. This scene may show limitations of the engine in its current state of implementation. We are looking forward to seeing your experiments! Remark: Feel free to take over pieces of code from other scenes.

*Good luck,*

Roland & Oliver