



COMP220: Graphics & Simulation
10: Simulation & Animation

Worksheet Schedule

Worksheet	Start	Formative deadline
1: Framework	Week 2	Mon 15th Feb 4pm (Week 4)
2: Basic scene	Week 4	Mon 1st Mar 4pm (Week 6)
3: Plan/prototype	Week 6	Mon 15th Mar 4pm (Week 8)
4: Final iteration	Week 8	Mon 12th Apr 4pm (Week 10)

Learning outcomes

By the end of this week, you should be able to:

- ▶ **Recall** the key concepts involved in solving mechanics problems
- ▶ **Write** programs which feature realistic physics simulations
- ▶ **Describe** how a rigged model is transformed to produce animation

Agenda

Agenda

- ▶ Lecture (async):
 - ▶ **Recap** the mathematical concepts of Newtonian physics.
 - ▶ **Explain** the role of rigging in 3D animation.
 - ▶ **Introduce** the scene graph and techniques for enhancing character animation.

Agenda

- ▶ Lecture (async):
 - ▶ **Recap** the mathematical concepts of Newtonian physics.
 - ▶ **Explain** the role of rigging in 3D animation.
 - ▶ **Introduce** the scene graph and techniques for enhancing character animation.
- ▶ Workshop (sync):
 - ▶ **Include** the Bullet Physics library in the OpenGL application.
 - ▶ **Explore** physics and animation techniques in code.

Schedule

16:00-16:10	Arrival, sign-in & overview
16:10-17:00	Demo & Exercise: Adding Bullet to your application
17:00-18:00	Demo & Exercise: Fun with Physics

The Bullet Physics Engine



Bullet

Bullet

- ▶ Bullet is an open-source Physics engine for simulation of rigid bodies (and more)

Bullet

- ▶ Bullet is an open-source Physics engine for simulation of rigid bodies (and more)
- ▶ Created by Erwin Coumans, who previously worked on the [Havok](#) engine

Bullet

- ▶ Bullet is an open-source Physics engine for simulation of rigid bodies (and more)
- ▶ Created by Erwin Coumans, who previously worked on the [Havok](#) engine
- ▶ **API** allows setting up a world and adding objects (and constraints) to it

Bullet

- ▶ Bullet is an open-source Physics engine for simulation of rigid bodies (and more)
- ▶ Created by Erwin Coumans, who previously worked on the [Havok](#) engine
- ▶ **API** allows setting up a world and adding objects (and constraints) to it
- ▶ Doesn't automatically update rendered objects - need to **manually send data** back and forth.

The Plan

We are going to:

The Plan

We are going to:

- ▶ See where to [download](#) the Bullet source code from and [how to compile](#) it

The Plan

We are going to:

- ▶ See where to [download](#) the Bullet source code from and [how to compile](#) it
- ▶ Integrate Bullet with our OpenGL application

The Plan

We are going to:

- ▶ See where to [download](#) the Bullet source code from and [how to compile](#) it
- ▶ Integrate Bullet with our OpenGL application
- ▶ Look at some of the features of the physics engine, including the debug drawing functionality and how it can be used