

## CODE ISCG6442 Course Name: Game Programming

Course number: Level: 6 Credits: 15 credits  
 Main programme: BCS Delivery:  
 Endorsement: Hours directed: 39  
 Other programmes: GDCMP Hours self-directed: 111  
 Prerequisites: ISCG5421 for BCS only Total hours: 150  
 Co-requisites: ISCG6428 Number of weeks: 16 weeks  
 Restrictions: None  
 Entry requirements: BCS & GDCMP

Students are expected to adhere to United's policy on conduct in respect of staff, fellow students, and in the use of resources and facilities.

### NZQA Level Descriptor:

	<b>Knowledge</b>	<b>Skills</b>	<b>Application</b>
6	<i>Specialised technical or theoretical knowledge with depth in one or more fields of work or study</i>	<i>Analyse, generate solutions to unfamiliar and sometimes complex problems.</i>  <i>Select, adapt and apply a range of processes relevant to the field of work or study</i>	<i>Advanced generic skills and/or specialist knowledge and skills in a professional context or field of study</i>

### Course aim:

To provide students with programming knowledge and mathematical concepts required for effective game development.

### Learning outcomes:

1. Utilise a collection of physics and mathematical concepts for a game.
2. Develop a simple text-based game.
3. Develop a sprite-based game using widely accepted techniques, including physics, mathematics and programming concepts

### Topics may include:

Programming Concepts for Games, Physics Concepts for Games, and Mathematical Concepts for Games, Logic and Decision Making for Games

Expanded Outcomes
<b>Outcome 1: Utilise a collection of physics and mathematical concepts for a game.</b> <ul style="list-style-type: none"> <li>• Understand the theory and concept of Cartesian geometry.</li> <li>• Use translation, rotation, scaling and skewing for game solutions.</li> <li>• Use trigonometry and simple rotational mathematics to solve simple and complex game problems.</li> <li>• Use vector-based and matrix-based mathematics to solve simple and complex game problems.</li> <li>• Use sets and logic to solve basic problems.</li> <li>• Understand basic Newtonian physics and its application in games.</li> </ul>
<b>Outcome 2: Develop a simple text-based game.</b> <ul style="list-style-type: none"> <li>• Utilise widely accepted good programming practices.</li> <li>• Use basic input/output methods.</li> <li>• Use effective game logic.</li> <li>• Detect the win/lose/tie condition.</li> </ul>
<b>Outcome 3: Develop a sprite-based game using widely accepted techniques, including physics,</b>