

## Module Specification

### Module Summary Information

<b>1</b>	<b>Module Title</b>	Computer Programming
<b>2</b>	<b>Module Credits</b>	20
<b>3</b>	<b>Module Level</b>	4
<b>4</b>	<b>Module Code</b>	CMP4266

<b>5</b>	<b>Module Overview</b>
<p>Within the module of Computer Programming you will learn the key skills of Programming and how this relates to technology and communications. Programming is an engineering tool that plays a vital role to drive most of the modern technologies surrounding us, including the technological devices for communication, transportation and entertainment. In other words, it can be said that our modern lifestyles are heavily dependent on programming. Moreover, businesses increasingly rely on computers and the software run on them. Programming skills and a broader and deeper understanding of programming are therefore becoming increasingly important to the jobs market.</p> <p>This introductory computer programming module provides the underpinning knowledge and practice for computing students to design, build and test software components. The module will make use of practical sessions primarily to allow you to apply programming principles and constructs in order to creatively solve problems by means of developing small programs.</p> <p>Module content and assessment enables learners to acquire programming skills in a modern imperative language.</p>	

<b>6</b>	<b>Indicative Content</b>
<p>Topics that will be covered include:</p> <ul style="list-style-type: none"> <li>• Input/Output</li> <li>• Built-in Data Structures</li> <li>• Iterations</li> <li>• Functions and Parameters</li> <li>• Objects and Classes</li> <li>• Documentation</li> <li>• Unit Testing</li> <li>• Graphical User Interfaces</li> <li>• Events and event handling</li> </ul>	

<b>7</b>	<b>Module Learning Outcomes</b>	
	<b>On successful completion of the module, students will be able to:</b>	
	<b>1</b>	Apply problem solving skills in order to design solutions to programming problems.
	<b>2</b>	Implement those solutions in an imperative programming language by using common programming tools (such as editors and interpreters).
	<b>3</b>	Use common programming tools and techniques (e.g. IDE, testing APIs and theories) to test, evaluate programs and document findings effectively in appropriate format.

<b>8</b>	<b>Module Assessment</b>		
<b>Learning Outcome</b>			
	<b>Coursework</b>	<b>Exam</b>	<b>In-Person</b>
<b>1-3</b>	<b>X</b>		

<b>9</b>	<b>Breakdown Learning and Teaching Activities</b>	
<b>Learning Activities</b>		<b>Hours</b>
<b>Scheduled Learning (SL)</b> includes lectures, practical classes and workshops, peer group learning, Graduate+, as specified in timetable		48
<b>Directed Learning (DL)</b> includes placements, work-based learning, external visits, on-line activity, Graduate+, peer learning, as directed on VLE		90
<b>Private Study (PS)</b> includes preparation for exams		62
<b>Total Study Hours:</b>		200