

## MODULE DESCRIPTOR

### Module Title

Computer Architecture and Programming

Reference	EN1542	Version	2
Created	November 2023	SCQF Level	SCQF 7
Approved	June 2021	SCQF Points	15
Amended	April 2024	ECTS Points	7.5

### Aims of Module

To equip students with the skills to articulate computer architecture concepts and proficiently design, test, and implement software programs in high-level languages to address engineering challenges.

### Learning Outcomes for Module

On completion of this module, students are expected to be able to:

- 1 Describe foundational principles of computer architecture and hardware.
- 2 Implement modular structured programs in a high-level programming language to prescribed standards and specifications.
- 3 Define the characteristics of a typical programming language, algorithms and data structures and the process of software development.

### Indicative Module Content

Hardware: Fundamentals of computer architecture (such as Von Neumann architecture), instruction set, memory hierarchies, main memory organisation, cache organisation, parallelism, pipelining, and hazards. Software development: algorithms, source and object code, compilers, the edit-compile-execute cycle, software design, testing, standards and documentation. Syntax of a high-level language: constants and variables, data types, pointers, arrays and data structures; program expressions and statements, input and output, selection and repetition control structures; modular programming, library and user functions, parameter passing, macros.

### Module Delivery

This module is taught using a structured programme of lectures, tutorials and laboratory exercises supplemented by directed reading and student-centred learning.

Indicative Student Workload	Full Time	Part Time
Contact Hours	60	N/A
Non-Contact Hours	90	N/A
Placement/Work-Based Learning Experience [Notional] Hours	N/A	N/A
TOTAL	150	N/A
Actual Placement hours for professional, statutory or regulatory body		

## ASSESSMENT PLAN

If a major/minor model is used and box is ticked, % weightings below are indicative only.

### Component 1

Type:	Coursework	Weighting:	50%	Outcomes Assessed:	2, 3
Description:	Coursework of software development				

### Component 2

Type:	Examination	Weighting:	50%	Outcomes Assessed:	1
Description:	Closed book assessed tutorials or online quizzes.				

## MODULE PERFORMANCE DESCRIPTOR

### Explanatory Text

Each component comprises 50% of the module grade. To pass the module, a D grade is required.

		Coursework:						
		<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>	<b>NS</b>
Coursework:	<b>A</b>	A	A	B	B	E	E	
	<b>B</b>	A	B	B	C	E	E	
	<b>C</b>	B	B	C	C	E	E	
	<b>D</b>	B	C	C	D	E	E	
	<b>E</b>	E	E	E	E	E	F	
	<b>F</b>	E	E	E	E	F	F	
<b>NS</b>		Non-submission of work by published deadline or non-attendance for examination						

## Module Requirements

Prerequisites for Module	None.
Corequisites for module	None.
Precluded Modules	None.

## INDICATIVE BIBLIOGRAPHY

- 1 F. Wempen, "Computing Fundamentals: Introduction to Computers," Sybex, January 2015. ISBN: 9781119039716.
- 2 WEERT, P.V. and GREGOIRE, M., 2016. C++ standard library quick reference. Berkeley, CA: Apress.
- 3 HORTON, I., 2014. Beginning C++. Berkeley, CA: Apress.
- 4 SUTHERLAND, B., 2015. C++ recipes: a problem-solution approach. Berkeley, CA: Apress.