Module Title	**Fundamentals of Computer Science**
Level	4
Reference Code	CSI_4_FCS
Credits	20
Student Study Hours	Total: 200 Contact hours: 52 Student managed learning hours: 148
Pre-Requisites	None
Co-requisites	None
Excluded combinations	None
Module coordinator	TBC
Division	Division of Computer Science and Informatics
Short Description	The history of computing, the role and the science of algorithms, abstraction, the binary numbering system and its representations, Boolean logic and gates, fundamentals of computer hardware, building computer circuits, the von Neumann model, introduction to hardware, data, and software, the nature of data and its operation.
Aims	This module aims to give a comprehensive introduction to Computer Science, its theoretical basis and practical implementation in both hardware and software.
Learning Outcomes	**LO1: Knowledge and Understanding** ● Describe the historical foundations and contemporary development of computing theory and computer hardware. (Maps to: BCS 2.2.1 a1-a6, a9) **LO2: Intellectual Skills** ● Describe what theoretical computing is and how it has been mechanically implemented in physical computers. implementations (Maps to: BCS 2.2.1 a1-a6, a9) **LO3: Practical Skills** ● Work with binary numbers and Boolean logic and identify the principal functionalities of the components of computer hardware. (Maps to: BCS 2.2.1 a7-a8) **LO4: Transferable Skills** ● Reason about the underlying functional mechanisms at work in all kinds of computer systems. (Maps to: BCS 2.2.1 c1-c2)
Employability	This module provides an understanding of the basis on which all computer systems are built that gives immediate insight into their probable capabilities and limitations. This enhances your ability to quickly learn new systems whether as a user, administrator or developer. Almost every profession now involves computers and computing in some capacity, and this ability will bring practical benefits to all of them.
Teaching and Learning Pattern	The module will be delivered as a combination of lectures, student-driven discussions and practical tutorials in the labs.
Indicative Content	● The concept and definition of algorithms ● The history of computer architectures ● The von Neumann machine and the fetch, decode execute cycle ● Boolean logic and its implementation in hardware ● Binary numbers and their use in addressing and data storage ● Memory, caches, latency and storage mediums ● I/O fundamentals: handshaking, buffering, programmed I/O, interrupt-driven I/O

	● Fetch, decode and execute cycle ● Assembly language programming
Assessment *Elements &* *weightings*	**Coursework 100%** **Summative Assessment** The assessed coursework based on individual assessments is expected to take the form of in-class tests evenly spaced through the teaching. (LO1-LO4) **Formative Assessment** Skills for the summative assessment will be embedded throughout formative opportunities in Lectures and Workshops. Formative assessment will take different forms, such as: ● interactive revision quizzes ● verbal feedback on tutorial activities ● observation and questioning to provide instant feedback ● self-assessment
Indicative Sources *(Reading* *lists)*	**Core:** ● J. G. Brookshear, J., Brylow, D. (2019) Computer Science: an overview, Pearson Education, Limited **Optional:** ● Dell, N., Lewis, J. (2015) *Computer Science Illuminated* 6th edition. Jones and Bartlett Publishers, Inc. ● Forouzan, Behrouz A. (2014) *Foundations of Computer Science* 3rd edition. Andover: Cengage Learning, 2014.