UNIVERSITY OF LONDON

BSc Computer Science

Module Specification

Key Information			
Module title	Fundamentals of Computer Science		
Level	4	Credit value	15
Member Institution	Goldsmiths	Notional study hours and duration of course	150
Module lead author/ Subject matter expert			
Module co-author			

Rationale for the module

This module covers the basic underpinnings of computer science that provide a foundation for degree level study. It will give a broad overview of the field that will support learning in other modules and will be expanded further within the degree.

Aims of the module

By taking this module, you will gain a broad understanding of many of the key topic areas in computer science and the fundamental concepts that underpin them. In the area of fundamental concepts, you will study binary representations and logic, complexity theory and theories of computation, finite state machines and Turing machines. Building on this, you will then study key areas of interest in computer science including databases, artificial intelligence, and machine learning. These will be presented in the light of practical examples to illustrate how they are implemented in modern computer systems.

Topics covered in this module:

The topics listed here are an approximation of what will be covered. The topics presented may be slightly revised to ensure currency and relevance. Students will be advised of any changes in advance of their study.

1. Logic
2. Proof techniques
3. Basic combinatorial principles
4. Automata theory
5. Regular languages
6. Context-free languages
7. Turing machines
8. Algorithms 1
9. Algorithms 2
10. Complexity theory

Approximately 10-12 hours of study will be required per topic. The remaining study time is intended for coursework and examination preparation.

Learning outcomes for the module

Students who successfully complete this module will be able to:

1. Understand logical arguments and apply basic concepts of formal proof
2. Analyse and predict the behaviour of an algorithm using mathematical techniques
3. Understand the process of algorithmic thinking and a number of proof techniques and apply this knowledge to solve a range of computer science problems
4. Understand and apply various concepts in automata theory such as deterministic automata, regular languages, and context-free grammar
5. Understand the process of computation through Turing machines

Assessment strategy, assessment methods

Summative and Formative Assessments
The module will contain a range of summative and formative assessments. Summative assessments are assessments which contribute directly towards your final grade. Formative assessments do not count directly towards your final grade. Instead, they provide you with opportunities for low stakes practice, and will often provide some sort of feedback about your progress. For example, a practice quiz might provide you with feedback about why a particular answer was wrong.

Assessment Activities
The table below lists the assessment activity types you might encounter taking the module. It also states if that type of assessment can be automatically graded. For example, multiple choice quizzes can be automatically graded, and so can some programming assignments. It also states if that type of assessment will be found in the summative coursework and the summative examination. More details about the summative assessments are provided below.

Assessment activity type	Can it be automatically graded with feedback in some cases?	Coursework	Examination
Quiz	X	X	X
Writing task		X	X
Simulation task	X	X	
Peer review task		X	

Pass Mark

In order to pass this module, you must achieve at least 35% in each element of summative assessment and an overall weighted average of 40%, subject to the application of rules for compensation. Please refer to the programme regulations for more information.

Summative Assessment Elements

This is a module that has a significant amount of theory so is assessed as a theory-based module. This means that the summative assessment is composed of two elements, whose weightings are listed in the table below.

Summative Assessment Component	Percentage of final credit	Deadline
Coursework	50%	Mid session
Examination	50%	End of session

The coursework comprises a variety of practical exercises and quizzes which in total will take up to 25 hours of study time to complete. The examination will be two hours long, and consist of written answer and multiple choice questions.

Learning resources

The module will draw on a number of different, largely web-based, public resources as well as the resources produced as bespoke material for this module.

Specific essential readings from the following list are included in the Readings page for each week:

Kenneth H. Rosen (2011). Discrete Mathematics and its Applications, 7th. McGraw-Hill

Michael Sipser (2012). Introduction to the theory of computation, 3rd. Cengage Learning

John Hopcroft et al. (2013). Introduction to Automata Theory, Languages and Computation, Pearson