

FP096-30 Foundations of Computer Science

26/27

Department

Warwick Global Academy

Level

Foundation

Module leader

Joe Alcantara

Credit value

30

Module duration

25 weeks

Assessment

52% coursework, 48% exam

Study location

University of Warwick main campus, Coventry

Description

Introductory description

This module combines theoretical and practical aspects of computer science, introducing students to algorithms, programming, and the software development lifecycle. Students will study the role of formal languages and develop problem-solving skills by applying computing techniques across different contexts. Ethical, social, and commercial considerations in computing are also addressed. By the end of the module, students will be able to write and debug simple programs, apply computational thinking to solve problems, and reflect on the broader implications of computer science in society.

Module aims

This module combines the theory of computing with practical computing activities, including programming and application of the software lifecycle.

It aims to introduce students to the fundamental aspects of the academic discipline of Computer Science, illustrating the use of formal languages in computer science, including algorithms and programming.

It also aims to develop students' computing-related problem-solving skills whilst enabling them to apply computing skills to other areas and provides suitable preparation for higher education

courses in computing and related areas.

Outline syllabus

This is an indicative module outline only to give an indication of the sort of topics that may be covered. Actual sessions held may differ.

This is an indicative module outline only to give an indication of the sort of topics that may be covered. Actual sessions held may differ.

Block 1: Foundations of Computing

This block is a deep dive into the fundamental theory of how computers represent, process, and store information at a low level.

Block 2: The Programmer's Toolkit

This is a focused, hands-on block where students learn the craft of programming, from basic syntax to professional object-oriented design.

Block 3: Data Structures & Algorithms

With strong programming skills established, this block tackles the core of computer science: how to organise data efficiently and how to write algorithms to process it.

Block 4: The Wider Context

This final block connects their technical skills to the real world, covering the systems their code runs on and the professional and ethical responsibilities of a developer.

Learning outcomes

By the end of the module, students should be able to:

- Recognise and explain specific computing terminology, technology, key developments, principles and methods;
- Apply computational thinking to translate a range of problems into computer-based solutions;
- Use a combination of practical programming skills and theoretical knowledge to solve calculations, manipulate data and predict outcomes;
- Identify and describe the social, legal and ethical consequences of computer applications and technology;
- Display professional software engineering skills such as team working, time management and the importance of commercial reality; and
- Explain computing concepts, solutions and results to a range of audiences.

Indicative reading list

[Reading lists can be found in Talis](#)

International

All students are international, representing a wide range of cultural, linguistic, and educational

backgrounds. They bring diverse experiences and perspectives, enriching discussion and supporting the development of intercultural awareness. Examples for analysis and evaluation are drawn from a variety of countries, ensuring that teaching and learning are embedded in an international context.

Subject specific skills

Students will be able to:

interpret, analyse and explain computing principles and methods;
lead debates and draw conclusions from examining primary sources;
Develop computer-based solutions for a range of problems using software engineering skills.

Transferable skills

Communication and presentation skills;
Collaboration and group work skills;
Research skills;
Critical thinking.

Study

Study time

Type	Required
Seminars	25 sessions of 4 hours (33%)
Online learning (independent)	25 sessions of 1 hour (8%)
Private study	115 hours (38%)
Assessment	60 hours (20%)
Total	300 hours

Private study description

Extra-class reading of books and articles in preparation for lectures and seminars.
Home assignments.
Worksheets.
Flipped learning including pre-recorded videos.

Costs

No further costs have been identified for this module.

Assessment

You must pass all assessment components to pass the module.

Assessment group D

	Weighting	Study time	Eligible for self-certification
Assessment component			
Portfolio: Presentation	10%	6 hours	No
The first group task is the Project Proposal Presentation, where the team will formally outline the concept, scope, features, and development plan for a Python application they intend to build.			
Reassessment component is the same			
Assessment component			
Portfolio: Group Project	21%	12 hours	No
The core of the portfolio is the group Python Programming Project. This is the collaborative coding effort to build the proposed application, which must demonstrate mastery of key course concepts such as functions, classes, appropriate data structures, and algorithms.			
Reassessment component is the same			
Assessment component			
Portfolio: Reflection	21%	12 hours	Yes (extension)
Individual Reflective Journal. This is a significant component where students will document their thoughts on professional ethics, their personal learning journey in computer science, and the dynamics of their teamwork.			
Reassessment component is the same			
Assessment component			
Examination	30%	18 hours	No

Weighting

Study time

Eligible for self-certification

At the end of the module, there will be a formal written examination. This exam is designed to assess your individual understanding of the core theoretical concepts covered throughout the course. Topics will include, but are not limited to:

Number Systems and Binary Arithmetic
Boolean Algebra and Logic Circuits
Computer Architecture and Operating Systems
Data Structures and Algorithms
Software Engineering Principles and Ethics

-
- Answerbook Pink (12 page)
 - Students may use a calculator

Reassessment component is the same

Assessment component

Portfolio: Viva	18%	12 hours	No
-----------------	-----	----------	----

To ensure individual accountability, each student will participate in an Individual Viva. This is a one-to-one oral examination where they must explain and provide a rationale for a section of code they were primarily responsible for writing.

Reassessment component is the same

Feedback on assessment

Written Feedback on Tabula.

[Past exam papers for FP096](#)

Availability

There is currently no information about the courses for which this module is core or optional.