FP016-30 Computer Science

23/24

Department
Warwick Foundation Studies
Level
Foundation
Module leader
Leonardo Alves Dias
Credit value
30
Module duration
25 weeks
Assessment
57% coursework, 43% exam
Study location
University of Warwick main campus, Coventry

Description

Introductory description

FP016-30 Computer Science:
This module combines the theory of computing and practical activities to familiarise the student with the computer science field.

Module web page

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.

Part 1: Computer Architecture:
Computer Logic and Binary System.
Boolean Mathematics.
Boolean Circuit Design.
Computer Architectures.
Real Numbers.

Part 2: Computational Thinking and Python Programming:
Computational and Logical Thinking.
Python Programming Language.
Variables and Data Types.
Control flow structures: conditional and repetition.
Functions & Recursion.
Classes.
Python Programming Project.

Part 3: Ethics and Consequences:
Ethics of Computing: Data Security and Privacy.
Ethical, social and legal issues behind Computing Technologies Presentation.

Part 4: Computer Systems:
Hardware and Components.
Operating Systems.
Communication and Networks.
Data Transmission.

Part 5: Software Engineering and Business:
Introduction to Software Engineering.
Top-Down Design & Specification.
Implementation & Testing.
Deployment methodologies.

Part 6: Data Representation and Algorithms:
Numbers and Number Systems.
Data Representation and Transformation.
Algorithms for Searching and Sorting.
Data Structures.

Page 7: Object-Orientated Paradigm:
OOP Languages (Python).
Encapsulation.
Inheritance.
Polymorphism.

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

-> Dale, Nell, and John Lewis. Computer Science Illuminated, Jones & Bartlett Learning, LLC, 2019.

-> Floyd, Thomas. Digital Fundamentals, Global Edition, Pearson Education, Limited, 2015.

-> Karl Beecher. Computational Thinking: A Beginner's Guide to Problem-Solving and Programming. BCS, The Chartered Institute for IT, 2017.

-> Matthes, Eric. Python Crash Course, 2nd Edition: A Hands-On, Project-Based Introduction to Programming, No Starch Press, 2019.

-> Rob Mastrodomenico. The Python Book. Wiley, 2022.

-> Shaw, Zed A. Learn Python the hard way: a very simple introduction to the terrifyingly beautiful world of computers and code. Pearson Addison Wesley, 2017.

-> Brookshear, J., et al. Computer Science: an Overview, Global Edition, Pearson Education, Limited, 2019.

-> Halsall, Fred. Introduction to data communications and computer networks. Electronic systems engineering series, 1985.

-> Thomas H. Cormen. Introduction to algorithms. 3rd ed. Massachusetts Institute of Technology, 2009. ISBN 978-0-262-03384-8.

View reading list on Talis Aspire

Interdisciplinary

The Ethics Presentation (Summative assessment) includes research skills learned at EAP and IRS, such as academic citation and referencing.
Interdisciplinary links with mathematics.

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
Lectures	25 sessions of 1 hour (8%)
Seminars	75 sessions of 1 hour (25%)
Private study	140 hours (47%)
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.

Costs

No further costs have been identified for this module.

Assessment

You must pass all assessment components to pass the module.

Assessment group D4

	Weighting	Study time
Individual Presentation about Ethical issues.	17%	10 hours

	Weighting	Study time

The students must create and record a presentation to demonstrate their ability to identify, analyse, and explain the ethical, legal, and social issues that can arise from using computer-based technology.
This assessment will be marked individually.

Python Programming Project.	40%	24 hours

The students must create a piece of software using the Python Programming Language and deliver the following:

- A Python Project (.py or .ipynb files) with the solution to a given problem. This will be developed as a collaborative group.
- Write an individual reflective piece evaluating and reflecting on their contribution. This will be individually written by each group member.

The Python code must emphasise the programming techniques, the algorithms, and the data manipulation and structures applied. Meanwhile, the reflective piece of writing enables each individual member to show the software engineering process and display the skills learned in the module whilst evaluating the project.
The Python code will have a collective mark defined for the group, while the essay will be individually marked.

Theory Examination Summer (Weeks 4 to 9)	43%	26 hours

The students must be able to identify, analyse, recognise, and explain computer terminology, principles and methods related to computer science. Besides, students must demonstrate their ability to use different methodologies, concepts, and development techniques to solve various computer-related problems.

- Answerbook Gold (24 page)
- Students may use a calculator

Feedback on assessment

Written Feedback on Tabula.

Past exam papers for FP016

Availability

Courses

This module is Core for:

- FIOE Warwick International Foundation Programme
    - Year 1 of FP18 Warwick International Foundation Programme - Computer Science
    - Year 1 of FP19 Warwick International Foundation Programme - Engineering

This module is Core option list A for:

- FIOE Warwick International Foundation Programme
    - Year 1 of FP13 Warwick International Foundation Programme - Mathematics and Economics
    - Year 1 of FP12 Warwick International Foundation Programme - Science and Engineering