

Presentation pattern *October to June*

Module description

This module aims to help our students become computational thinkers. Students will learn how to:

- apply general-purpose data structures and algorithmic techniques to solve computational problems.
- explain in a clear and succinct way how an algorithm or data structure works.
- analyse the complexity of algorithms to support design choices.
- write readable, tested, and documented Python functions and classes to implement algorithms and abstract data types.
- explain the limits of computation and its practical implications.

The module covers key ideas such as abstraction, recursion, and Big-Theta notation, and deals with algorithms and data structures for sorting, searching and optimisation. Students will learn by practical problem solving, using Python in Jupyter notebooks.

Person specification

The person specification for this module should be read in conjunction with the [generic person specification](#) for an associate lecturer at The Open University.

As well as meeting all the requirements set out in the generic person specification, you should:

- have a considerable understanding of algorithms and data structures and a good awareness of key concepts in computability and computational complexity theory
- have programming experience
- be enthusiastic and knowledgeable about computational thinking as a key skill for the 21st century.

It would be an advantage to have:

- familiarity with the Python programming language
- familiarity with Jupyter notebooks
- experience of teaching algorithms and data structures
- experience of teaching computability and computational complexity theory
- experience of using electronic forms of distance learning teaching and support (but training is provided if needed).

Additional information

- You will mark three tutor-marked assignments (TMAs).
- You will be required to provide up to 10 tutorial hours, which could be either face-to-face or online.

Credits awarded to the student for the successful completion of a module:	30
Number of students in a standard group:	20
Salary band:	3
Average number of hours per teaching week:	3.5