# Module Specification

## Programming for Data Science

Version: 2023-24, v1.0, 08 May 2024

**Contents**

## Part 1: Information

**Module title:** Programming for Data Science

**Module code:** UFCE8V-15-M

**Level:** Level 7

**For implementation from:** 2023-24

**UWE credit rating:** 15

**ECTS credit rating:** 7.5

**College:** College of Arts, Technology and Environment

**School:** CATE School of Computing and Creative Technologies

**Partner institutions:** None

**Field:** Computer Science and Creative Technologies

**Module type:** Module

**Pre-requisites:** None

**Excluded combinations:** None

**Co-requisites:** None

**Continuing professional development:** No

**Professional, statutory or regulatory body requirements:** None

## Part 2: Description

**Overview:** This module introduces you to principles of data science and programming with a range of applications to appropriate domains.

**Features:** Not applicable

**Educational aims:** This module is intended for students with little or no programming experience but also for those keen to improve their programming skills. It aims to provide students with an understanding of the role computation can play in

solving problems and, regardless of their background, feel justifiably confident of their ability to write small programs that allow them to accomplish useful goals.

**Outline syllabus:** The module will cover the following topics:

Introduction:

- Introduction to the field of data science
- Developing a working knowledge of common development tools used in the context of data science (such as Jupyter Notebooks, Visual Studio Code and Git)

Basic Programming and Algorithm Design:

- Principles of programming through the Python programming language (such as variables, flow control and data structures)
- Use of appropriate IDE or an approved virtual programming environment for the practical sessions.
- Procedural implementation of statistical algorithms

Data analytics practices:

- Introduction to popular data analysis and statistical learning tools and frameworks (such as Pandas, Numpy and Matplotlib)
- Develop an understanding the concept of data structures (such as Lists, Dictionaries, Sets, Arrays and DataFrames).
- Learn how to read in data into Pandas' Data Frames, how to query these structures, how to generate summary tables, data groupings, and how to clean and manipulate data
- Develop skills in Data Visualisation to support numerical data analysis

## Part 3: Teaching and learning methods

**Teaching and learning methods:** The module, presented via our online virtual learning environment, consists of a clearly signposted, easy-to-navigate student journey through carefully chosen learning materials which are designed to engage and challenge students as they work towards achieving the module learning outcomes. Content may be in a range of formats, including clear well-written text, diagrams, animations, video and interactive video, activities, quizzes, asynchronous discussions, coding and interpretation exercises.

Students will be provided with as many opportunities as possible to 'perform their understanding' rather than just reading or watching video to passively acquire knowledge. This may be in the form of simple tasks, activities or quizzes that students can engage with in the online environment, or larger pieces of work that may require additional thought. Whatever their nature, such tasks will be authentic (connected to the real world) and directly relevant to the programme learning outcomes.

The online environment also provides important opportunities to encourage students to work with, and learn from, their peers. The careful use of structured online discussion forums helps to foster an active learning community and enable students to share their responses to key questions, and to discuss, and even challenge, each other's ideas.

All learning materials are produced and presented in a way that ensures that they are appropriate for as diverse an audience as possible. We follow W3C accessibility standards and ensure that all content can be used with all popular screen-readers, offering alternative formats where possible. In general, we aim to avoid using language, idioms, images or other devices which root the content in any particular culture or creed that instead adequately reflect the diversity of the student audience.

In general, modules are designed with a number of key learning principles in mind that align closely with those of the university.

**Module Learning outcomes:** On successful completion of this module students will achieve the following learning outcomes.

**MO1** Apply the principles of programming and data management to solve problems

**MO2** Demonstrate the use of an object-oriented paradigm when solving software problems

**MO3** Design and implement algorithms for numerical analysis

**MO4** Demonstrate the use of proactive error handling techniques to address software reliability and program vulnerability issues

**MO5** Critique and reflect on alternative solutions to a given problem or on their own work in a constructive way

**MO6** Undertake independent research activities with relation to innovative approaches to data science problem solving

**MO7** Demonstrate the use of Data Visualisation techniques for supporting numerical data analysis

**MO8** Demonstrate the use of a version control system (such as Git) as part of an integrated development process

**Hours to be allocated:** 150

**Contact hours:**

Independent study/self-guided study = 114 hours

E-learning/online learning = 36 hours

Total = 150

**Reading list:** The reading list for this module can be accessed at readinglists.uwe.ac.uk via the following link https://uwe.rl.talis.com/modules/ufcfvq-15-m.html

## Part 4: Assessment

**Assessment strategy:** The assessment strategy for this module is a single coursework assignment, in which students are expected to demonstrate their code solutions. On-going feedback will be provided, which enriches and deepens the

learning of the student. Summative assessment is achieved through the demonstration of a set of programming exercises solutions together with the submission of a reflective development process report. The reflective report should focus on the process taken to develop solutions. In particular, identifying any strengths/weaknesses in their approach, suggestions for improvements, and any alternative approaches that could have been taken.

Resit Assessment Strategy: Students will be required to submit a single coursework assignment with the same structure as described above, i.e. a set of programming exercises and a reflective development process report.

**Assessment tasks:**

**Practical Skills Assessment** (First Sit)
Description: Practical coursework
Weighting: 100 %
Final assessment: No
Group work: No
Learning outcomes tested: MO1, MO2, MO3, MO4, MO5, MO6, MO7, MO8

**Practical Skills Assessment** (Resit)
Description: Practical coursework
Weighting: 100 %
Final assessment: No
Group work: No
Learning outcomes tested: MO1, MO2, MO3, MO4, MO5, MO6, MO7, MO8

## Part 5: Contributes towards

This module contributes towards the following programmes of study:

Data Science [UWE online] MSc 2023-24