Prifysgol Wrecsam
Wrexham University

Module specification

When printed this becomes an uncontrolled document. Please access the Module Directory for the most up to date version by clicking on the following link: Module directory

Module Code	COM479
Module Title	Fundamentals of Machine Learning
Level	4
Credit value	20
Faculty	FACE
HECoS Code	100992
Cost Code	GACP
Pre-requisite module	N/A

Programmes in which module to be offered

Programme title	Core/Optional/Standalone
BSc (Hons) Computer Science	Core
BSc (Hons) Computer Science with Industrial Placement)	Core
Stand-alone module aligned to BSC (Hons) Computer Science for QA and assessment	Option

Breakdown of module hours

Learning and teaching hours	12 hrs
Placement tutor support hours	0 hrs
Supervised learning hours e.g. practical classes, workshops	24 hrs
Project supervision hours	0 hrs
Active learning and teaching hours total	**36 hrs**
Placement hours	0 hrs
Guided independent study hours	164 hrs
Module duration (Total hours)	**200 hrs**

Breakdown of module hours (to apply to Hong Kong Institute of Technology only)

Learning and teaching hours	12 hrs
Placement tutor support hours	0 hrs
Supervised learning hours e.g. practical classes, workshops	61 hrs

Project supervision hours	0 hrs
Active learning and teaching hours total	**73** hrs
Placement hours	0 hrs
Guided independent study hours	127 hrs
Module duration (Total hours)	200 hrs

Module aims

This module aims to introduce the fundamentals of programming through a relevant programming language that aligns to contemporary practice. This module will deliver content through the context of student subject areas and explore programming logic and contextual problem solving. Throughout this module, the concepts of programming will be tightly linked to the context of developing within an Integrated Development Environment (IDE). On completion of this module students should have a clear concept of the role of programming syntax and logic in the context of their chosen subject area. They should be able to utilise fundamental programming techniques and strategies to solve contextual problems or development criteria.

Module Learning Outcomes

At the end of this module, students will be able to:

1	Identify syntax and structure of an industry-standard programming language.
2	Relate programming concepts to the context of subject area.
3	Apply programming techniques within an Integrated Development Environment.
4	Demonstrate design solutions to contextualised problems.

Assessment

Indicative Assessment Tasks:
This section outlines the type of assessment task the student will be expected to complete as part of the module. More details will be made available in the relevant academic year module handbook.

Indicatively these portfolio pieces may assess initial threshold concepts for programming syntax near the start of the module to ensure progress. Later these coursework pieces could represent more complex and time-consuming contextualised problems or development criteria that require critical thought to apply a combination of techniques and strategies

Portfolio assessments may comprise multiple pieces of work that collectively demonstrate a student's knowledge and skills developed throughout the module. These may take the form of one or two substantial tasks, or a series of smaller tasks, typically ranging from one to eight across the duration of the module.

Assessment number	Learning Outcomes to be met	Type of assessment	Duration/Word Count	Weighting (%)	Alternative assessment, if applicable
1	1,2,3,4	Portfolio	4000 Words or Equivalent	100%	

Derogations

N/A

Learning and Teaching Strategies

In line with the Active Learning Framework, this module will be blended digitally with both a VLE and online community. This content will also be fully represented online through VLE content and will be available to students as they progress with the module. Indicatively these could be sections of code/diagrams/PDF tutorials/pre-recorded videos depending on what it most appropriate.

As this module progresses, the strategies will change to best support a diverse learning environment. Initially, the module will start with a heavier reliance on engaging tutor-led lectures, demonstrations, and workshops to ensure that the students get the relevant threshold concepts. As the module continues experiential and peer learning strategies will be encouraged as the students' progress with their coursework.

Assessment will occur throughout the module to solidify key threshold concepts of programming fundamentals. Individual briefs will be given to clearly identify areas of focus through each stage, so students are fully aware of their progress throughout.

Welsh Elements

This module is designed to support Welsh-speaking students in line with the Welsh Language Standards. While the primary delivery will be in English, students will have the opportunity to submit assessments, including coursework and projects, in Welsh if preferred. Relevant module materials, such as reading lists, key texts, and guidance, will be available bilingually upon request, ensuring accessibility for all students. Additionally, where possible, guest speakers, case studies, or examples may include references to the Welsh business context, especially in areas such as data use in local industries and Welsh public sector organisations.

The department encourages students to develop bilingual digital skills by incorporating Welsh-language datasets, tools, and resources where appropriate, offering an inclusive learning environment. We also support the development of bilingual visualisation techniques, enabling students to create digital outputs that reflect the Welsh language, should they wish to do so.

Indicative Syllabus Outline

Depending on the relevance to current industry trends a programming language will be chosen that aligns with contemporary practice. The following indicative syllabus outline will apply to many languages and contexts though terms, styles or order of delivery may vary:

- Conditionals & Loops

- Random Numbers
- Arrays & Sorting
- Saving & Loading
- Introduction to Data Structures
- Enumerated Data Types
- Methods & Functions
- Return Types, Parameters & Keywords
- Objects & Instances
- Console Interfaces & Simple Graphics
- Introduction to Source Control

Indicative Bibliography

Please note the essential reads and other indicative reading are subject to annual review and update.

Essential Reads:

- McGrath, M. (2020), *C# Programming in Easy Steps*, Second Edition, Warwickshire: In Easy Steps.

Other indicative reading:

- Dawson, M. (2014), *Beginning C++ Through Game Programming*, Fourth Edition, Boston: Cengage Learning. Ferrone, H. (2020), *Learning C# by Developing Games with Unity 2020,* Fifth Edition, Birmingham: Packt Publishing.
- Lutz, M. (2013), *Learning Python: Powerful Object-Oriented Programming*, Fifth Edition, California: O'Reilly Media.
- Shaw, Z. (2017), *Learn Python 3: The Hard Way*, Boston: Addison-Wesley.

Administrative Information

For office use only	
Initial approval date	10th May 2023
With effect from date	September 2026
Date and details of revision	08/11/2023 - Addition of programme titles during Computing revalidation 23/02/2026 – Addition of BSc Computing for Business following validation. 12/05/2026 Addition of specific contact hour requirement for partner Hong Kong Institute of Technology (HKIT) AM1
Version number	3