University of the West of Scotland

Module Descriptor

Session: 2024/25

Title of Module: Computational Methods			
Code: MATH07008	**SCQF Level: 7 (Scottish Credit and Qualifications Framework)**	**Credit Points: 20**	**ECTS: 10 (European Credit Transfer Scheme)**
School:	School of Computing, Engineering & Physical Sciences		
Module Co-ordinator:	Dr Wan Mekwi		

Summary of Module

This module will teach students the fundamentals of programming in Python. It is suitable for students of all STEM fields.

Students will learn to write basic computer programs to solve real-world STEM problems. They will learn how to generate, manipulate, analyse and visualise data. As well as programming and simulation skills, students will develop an understanding of how problems can be simulated and/or solved within a computer-based numerical environment.

Fundamentals of programming will be explored including using conditional statements and looping, modularising code via user-defined functions, and creating and manipulating mathematical functions.

Students will also be introduced to the concept of cybersecurity and its importance in storing, protecting and sharing professional and personal data.

This module will work to develop a number of the key 'I am UWS' Graduate Attributes to make those who complete this module:
Universal:
- academic: analytical, critical thinker, inquiring
- professional: collaborative
Work-ready:
- academic: Knowledgeable, digitally literate
- personal: motivated
- professional: enterprising
Successful:
- academic: autonomous, incisive, innovative
- personal: creative, imaginative, resilient
- professional: driven, daring

Module Delivery Method

Face-To-Face	Blended	Fully Online	HybridC	Hybrid 0	Work-Based Learning
☒	☐	☐	☐	☐	☐

See Guidance Note for details.

Campus(es) for Module Delivery						
The module will **normally** be offered on the following campuses / or by Distance/Online Learning: (Provided viable student numbers permit) (tick as appropriate)						
Paisley:	Ayr:	Dumfries:	Lanarkshire:	London:	Distance/Online Learning:	Other:
☒	☐	☐	☐	☐	☐	Add name

Term(s) for Module Delivery					
(Provided viable student numbers permit).					
Term 1	☐	Term 2	☒	Term 3	☐

Learning Outcomes: (maximum of 5 statements) **These should take cognisance of the SCQF level descriptors and be at the appropriate level for the module.** At the end of this module the student will be able to:	
L1	Develop computer algorithms and programs to solve STEM problems
L2	Generate, manipulate, analyse and visualise programming and simulation data
L3	Generate coherent reports based on output of computer programs
L4	Demonstrate how security measures are used to protect data, networks and software

Employability Skills and Personal Development Planning (PDP) Skills	
SCQF Headings	During completion of this module, there will be an opportunity to achieve core skills in:
Knowledge and Understanding (K and U)	SCQF Level **7** Basic knowledge of the structure of a high-level language. Structuring problems in a suitable format for program development and presentation. General knowledge and understanding of security measures are used to protect data, networks and software.
Practice: Applied Knowledge and Understanding	SCQF Level **7** Develop limited computer applications using a high-level language. Present information using commonly available libraries. Demonstrate how security measures are used to protect data, networks and software.

	Select and critically evaluate technical literature and other sources of information to solve complex problems. Through the development of algorithms, programs and simulations, apply an integrated or systems approach to the solution of complex problems. Use practical computer laboratory skills to investigate complex problems. Adopt a holistic and proportionate approach to the mitigation of security risks.
Generic Cognitive skills	SCQF Level **7** Use appropriate quantitative science and engineering tools to gather data in an appropriate format. Use logical and analytical skills to formulate and understand computer programs in a high-level scripting language.
Communication, ICT and Numeracy Skills	SCQF Level **7** Demonstrate the ability to communicate ideas and concepts through the use of presentation software. Demonstrate an understanding of the computer techniques available to enhance the communication of science and engineering ideas and concepts. Use computers to automate repetitive tasks. Use a high-level programming language to enhance the effectiveness of a computer in the context of modern work tasks.
Autonomy, Accountability and Working with others	SCQF Level **7** Develop an enhanced level of transferable skills that will be of value in working with others in more complex situations. Function effectively as an individual and as a member or leader of a team. Be able to demonstrate the effectiveness of the team's performance and individual contribution.

Pre-requisites:	Before undertaking this module, the student should have undertaken the following:	
	Module Code:	**Module Title:**
	Other:	
Co-requisites	**Module Code:**	**Module Title:**

*Indicates that module descriptor is not published.

Learning and Teaching	
In line with current learning and teaching principles, a 20-credit module includes 200 learning hours, normally including a minimum of 36 contact hours and maximum of 48 contact hours.	
Learning Activities During completion of this module, the learning activities undertaken to achieve the module learning outcomes are stated below:	**Student Learning Hours** (Normally totalling 200 hours): (Note: Learning hours include both contact hours

	and hours spent on other learning activities)
Lecture/Core Content Delivery	12
Laboratory/Practical Demonstration/Workshop	24
Independent Study	164
	Hours Total 200

****Indicative Resources: (e.g. Core text, journals, internet access)**

The following materials form essential underpinning for the module content and ultimately for the learning outcomes:

Class notes as published on the University VLE.

Two free textbook resources available below:

"A practical Introduction to Python Programming" – B. Heinold
"Introduction to Scientific Programming with Python" – J. Sundnes

Please ensure the list is kept short and current. Essential resources should be included, broader resources should be kept for module handbooks / Aula VLE.

Resources should be listed in Right Harvard referencing style or agreed professional body deviation and in alphabetical order.

(**N.B. Although reading lists should include current publications, students are advised (particularly for material marked with an asterisk*) to wait until the start of session for confirmation of the most up-to-date material)

Attendance and Engagement Requirements

In line with the Student Attendance and Engagement Procedure: Students are academically engaged if they are regularly attending and participating in timetabled on-campus and online teaching sessions, asynchronous online learning activities, course-related learning resources, and complete assessments and submit these on time.

Equality and Diversity

The University's Equality, Diversity and Human Rights Procedure can be accessed at the following link: UWS Equality, Diversity and Human Rights Code.

Please ensure any specific requirements are detailed in this section. Module Co-ordinators should consider the accessibility of their module for groups with protected characteristics..

(N.B. Every effort will be made by the University to accommodate any equality and diversity issues brought to the attention of the School)

Supplemental Information

Divisional Programme Board	Engineering and Physical Sciences
Assessment Results (Pass/Fail)	Yes ☐No ☒
School Assessment Board	Computing, Engineering and Physical Sciences
Moderator	Dr Kwok Chi Chim
External Examiner	P. Wilson
Accreditation Details	TBC
Changes/Version Number	1.09. Change of title. Change to module summary. Change to student learning hours. Change to indicative resources. Change to Assessment Outcomes and Portfolio

Assessment: (also refer to Assessment Outcomes Grids below)
The module is assessed by a series of coursework exercises, forming one component, and a final unseen exercise forming a second component.
Assessment 1: A portfolio of practical work (a series of programming tasks) (50%)
Assessment 2: A portfolio of practical work (applications to engineering, science or data analysis) (50%)
(N.B. (i) **Assessment Outcomes Grids** for the module (one for each component) can be found below which clearly demonstrate how the learning outcomes of the module will be assessed. (ii) An **indicative schedule** listing approximate times within the academic calendar when assessment is likely to feature will be provided within the Student Module Handbook.)

Assessment Outcome Grids (See Guidance Note)

Component 1						
Assessment Type (Footnote B.)	**Learning Outcome (1)**	**Learning Outcome (2)**	**Learning Outcome (3)**	**Learning Outcome (4)**	**Weighting (%) of Assessment Element**	**Timetabled Contact Hours**
Portfolio of practical work	✔	✔	✔	✔	50	0

Component 2						
Assessment Type (Footnote B.)	**Learning Outcome (1)**	**Learning Outcome (2)**	**Learning Outcome (3)**	**Learning Outcome (4)**	**Weighting (%) of Assessment Element**	**Timetabled Contact Hours**
Portfolio of practical work	✔	✔	✔	✔	50	0